



การพัฒนาและสนับสนุนระบบ Payroll
DEVELOPMENT AND SUPPORT FOR PAYROLL SYSTEM

นางสาวณัฐกฤตา อภิรมย์เกษม

โครงการสหกิจศึกษานี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
ปริญญาวิทยาศาสตรบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีไทย-ญี่ปุ่น

พ.ศ. 2561

การพัฒนาและสนับสนุนระบบ Payroll
DEVELOPMENT AND SUPPORT FOR PAYROLL SYSTEM

นางสาวณัฐกฤตา อภิรมย์เกษม

โครงการสหกิจศึกษานี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
วิทยาศาสตรบัณฑิต สาขาเทคโนโลยีสารสนเทศ
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีไทย - ญี่ปุ่น
ปีการศึกษา 2561

คณะกรรมการสอบ

..... ประธานกรรมการสอบ

(ผศ. ดร.ประจักษ์ เจ็ดโคม)

..... กรรมการสอบ

(ผศ. ดร.นรังสรรค์ วิไลสกุลยง)

..... อาจารย์ที่ปรึกษา

(อาจารย์อดิศักดิ์ เลือสมิง)

..... ประธานสหกิจศึกษาสาขาวิชา

(อาจารย์สลิล ชีวกิตการ)

ลิขสิทธิ์ของสถาบันเทคโนโลยีไทย - ญี่ปุ่น

ชื่อโครงการ	การพัฒนาและสนับสนุนระบบPayroll
	Development and Development for Payroll System
ผู้เขียน	นางสาวณัฐกฤตา อภิรมย์เกษม
คณะวิชา	เทคโนโลยีสารสนเทศ สาขาวิชา เทคโนโลยีสารสนเทศ
อาจารย์ที่ปรึกษา	อาจารย์อดิศักดิ์ เสือสมิง
พนักงานที่ปรึกษา	นางสาวภัทรทา หมั่นเมือง
ชื่อบริษัท	บริษัท เอ-โฮสต์ จำกัด
ประเภทธุรกิจ/สินค้า	ให้บริการทางด้าน Hosting Service และ IBM & Oracle Product

บทสรุป

ในสหกิจศึกษาได้รับมอบหมายในตำแหน่ง Programmer ช่วยพัฒนาระบบ Payroll แก้ไขปัญหาต่างๆด้านการจัดการระบบเงินเดือน คำนวณเงินล่วงเวลา คำนวณภาษี ประกันสังคม กองทุนสำรองเลี้ยงชีพ ฯลฯ เพื่อให้บรรลุวัตถุประสงค์ขององค์กร ให้มีความถูกต้องแม่นยำ ความรวดเร็ว โดยสามารถแบ่งรูปแบบองค์กรเป็นสาขา โครงการ แผนงาน แผนก ฝ่าย และแบ่งตำแหน่งงาน หรือระดับพนักงานได้ สามารถกำหนดเงินได้ เงินหัก โดยใช้ระบบPayroll ที่พัฒนาเข้ามาช่วยจัดการ

จากการพัฒนาระบบPayroll ในส่วนของ การลา พนักงาน การลงเวลา ฯลฯ และ แก้ไข Issue ต่างๆ ที่ได้รับมอบหมายให้ระบบสามารถใช้งานได้ตลอดเวลา ทำให้เรียนรู้การใช้งานระบบ Oracle Application Express (APEX) มากขึ้น อีกทั้งสามารถบริหารเวลาและอดทนกับสภาวะกดดัน สามารถเข้าใจถึงความรับผิดชอบในหน้าที่ที่ได้รับมอบหมาย ซึ่งสามารถนำไปใช้ในอนาคตภายภาคหน้า

กิตติกรรมประกาศ

ในการปฏิบัติงานที่บริษัท เอ-โฮสต์ จำกัด ได้รับความกรุณาและความช่วยเหลือและดูแลจากหลายๆบุคคล ได้เรียนรู้ประสบการณ์ทำงานจริง และเรียนรู้ต่างๆอีกมากมาย อีกทั้งคอยให้คำปรึกษาทำให้งานปฏิบัติครั้งนี้สำเร็จลุล่วงไปด้วยดี

ขอขอบคุณ คุณภัทรทา หมั่นเมือง ที่ได้เสียสละเวลาเป็นที่ปรึกษาในช่วงสหกิจและพี่ๆในบริษัททุกท่านที่คอยให้ ความช่วยเหลือ ทุกครั้งที่เกิดข้อสงสัยระหว่างสหกิจศึกษา รวมถึงข้อสงสัยต่างๆให้กระจ่าง และคำแนะนำเทคนิคในการทำงานที่สามารถนำไปใช้ในชีวิตจริงได้ ขอขอบพระคุณทุกท่าน ไว้ ณ ที่นี้ด้วย

นางสาวณัฐกฤตา อภิรมย์เกษม
ผู้จัดทำ

TNI

สารบัญ

บทสรุป	ก
กิตติกรรมประกาศ	ข
สารบัญ	ค
สารบัญรูป	ง
สารบัญตาราง	ฉ
บทที่ 1 บทนำ	1
1.1 ชื่อและที่ตั้งของสถานประกอบการ	1
1.2 ลักษณะธุรกิจของสถานประกอบการ หรือการให้บริการหลักขององค์กร	2
1.3 รูปแบบการจัดองค์กรและการบริหารองค์กร	6
1.4 ตำแหน่งและหน้าที่งานที่นักศึกษาได้รับมอบหมาย	7
1.5 พนักงานที่ปรึกษา และ ตำแหน่งของพนักงานที่ปรึกษา	7
1.6 ระยะเวลาที่ปฏิบัติงาน	8
1.7 ที่มาและความสำคัญของปัญหา	8
1.8 วัตถุประสงค์หรือจุดมุ่งหมายของโครงการ	8
1.9 ผลที่คาดว่าจะได้รับการปฏิบัติงานหรือโครงการที่ได้รับมอบหมาย	9
1.10 นิยามศัพท์เฉพาะ	9
บทที่ 2 ทฤษฎีและเทคโนโลยีที่ใช้ในการปฏิบัติงาน	10
2.1 ภาษา SQL (Structured Query Language)	10
2.1.1 ประเภทของคำสั่งของภาษาSQL	10
2.1.2 การเรียกค้นข้อมูลอย่างง่าย	15
2.1.3 ชนิดข้อมูลที่ใช้ในภาษา SQL	17
2.1.4 ตัวดำเนินการ (Operator)	18
2.1.5 ฟังก์ชันที่ใช้ในภาษา SQL	20
2.2 ภาษา PL/SQL (Procedural programming language)	22
2.2.1 โครงสร้างของ PL/SQL Block	23

สารบัญ(ต่อ)

2.2.2 Identifiersคือชื่อที่ตั้งให้แก่ PL/SQL Object ต่างๆดังต่อไปนี้	24
2.2.3 การประกาศตัวแปรและการใช้ตัวแปร	25
2.2.4 การใช้คำสั่ง SQL ภายในบล็อกของ PL/SQL	27
2.2.5 คำสั่งโครงสร้างการเลือก (Selection Statement).....	28
2.2.6 สั่งโครงสร้างการวนรอบ (Iteration)	29
2.3 JavaScript.....	32
2.3.1 คำสั่งการเขียนสคริปต์ของ Java Script เริ่มต้นด้วย.....	32
2.3.2 การประกาศตัวแปร(Declaring variables).....	32
2.3.3 การสร้าง function	33
2.3.4 ชนิดข้อมูลของตัวแปร	33
2.3.5 Reserved Words.....	33
2.4 Oracle Application Express (Oracle APEX)	34
2.4.1 การพัฒนาโปรแกรมแบบDeclarative	36
2.4.2 องค์ประกอบ APEX	36
2.5 ข้อมูลระบบเงินเดือน	37
2.5.1 เงินได้.....	37
2.5.2 เงินหัก	37
2.5.3 ค่าลดหย่อน	37
2.5.4 เงินล่วงเวลา.....	37
บทที่ 3 แผนงานการปฏิบัติงานและขั้นตอนการดำเนินงาน	38
3.1 แผนงานการฝึกงาน	38
3.2 รายละเอียดที่นักศึกษาปฏิบัติในการฝึกงาน	39
3.2.1 Oracle Application Express (APEX).....	39
3.2.2 งานอื่นๆ	39
3.3 ขั้นตอนการดำเนินงานที่นักศึกษาปฏิบัติงาน	39

สารบัญ(ต่อ)

3.3.1 ศึกษาการใช้งานแอปพลิเคชันที่จำเป็นต่อการพัฒนาระบบPayroll.....	39
3.3.2 การใช้งานแอปพลิเคชัน Oracle Application Express (APEX)	39
3.3.3 การสร้างฟังก์ชันและทำPOC.....	41
3.3.4 การเสริม Plug-in	42
3.3.5 ตรวจสอบความถูกต้องและแก้ไขปัญหา.....	42
3.3.6 จัดการทำเอกสารที่เกี่ยวข้อง	42
บทที่ 4 สรุปผลการดำเนินงาน การวิเคราะห์และสรุปผลต่าง ๆ.....	43
4.1 ขั้นตอนและผลการดำเนินงาน	43
4.1.1 พัฒนาระบบเงินเดือนด้วย Oracle Apex.....	43
4.1.2 POC เทคโนโลยีใหม่ ๆ มาใช้ใน Oracle Apex	94
4.1.3 ทดสอบและแก้ไขข้อผิดพลาดของระบบ	109
4.2 ผลการวิเคราะห์ข้อมูล	110
4.3 วิเคราะห์และวิจารณ์ข้อมูลโดยเปรียบเทียบผลที่ได้รับกับวัตถุประสงค์และจุดมุ่งหมายในการปฏิบัติงานหรือการจัดทำโครงการ	113
บทที่ 5 บทสรุปและข้อเสนอแนะ	115
5.1 สรุปผลการดำเนินงาน.....	115
5.2 แนวทางการแก้ไขปัญหา.....	115
5.3 ข้อเสนอแนะจากการดำเนินงาน.....	116
เอกสารอ้างอิง.....	117
ประวัติผู้จัดทำโครงการ	118

สารบัญรูป

รูป

หน้า

รูปที่ 1.1 แผนที่ บริษัท เอ-โฮสต์ จำกัด	1
รูปที่ 1.2 รางวัลที่บริษัท A-HOST ได้รับ.....	3
รูปที่ 1.3 คณะผู้บริหาร A-HOST Company Limited	6
รูปที่ 2.2 JavaScript Reserved Words	34
รูปที่ 2.3 Oracle Application Express (Oracle APEX).....	34
รูปที่ 2.4 APEX Architecture	35
รูปที่ 2.5 องค์ประกอบ APEX	36
รูปที่ 3.1 ตัวอย่างหน้าการใช้งาน Apex	40
รูปที่ 3.2 ตัวอย่างรายละเอียดการสร้างแอปพลิเคชัน	40
รูปที่ 3.3 ตัวอย่างรายละเอียดการสร้างหน้ารูปแบบต่างๆ.....	41
รูปที่ 3.4 ตัวอย่างการทดสอบเบื้องต้น(POC)	41
รูปที่ 3.5 ตัวอย่างหน้าดาวน์โหลด Plug-in	42
รูปที่ 3.6 ตัวอย่างการเทียบ Plug-in	42
รูปที่ 4.1 ขั้นตอนการเข้าหน้า Edit Page.....	44
รูปที่ 4.2 ขั้นตอนการสร้างRadio Group(1).....	44
รูปที่ 4.3 ขั้นตอนการสร้างRadio Group(2)	45
รูปที่ 4.4 ขั้นตอนการสร้างRadio Group(3).....	45
รูปที่ 4.5 ขั้นตอนการสร้างไอเทมรับค่าตัวแปร Radio Group (1).....	46
รูปที่ 4.6 ขั้นตอนการสร้างไอเทมรับค่าตัวแปร Radio Group (2).....	46
รูปที่ 4.7 ขั้นตอนการสร้างDynamic Action ในRadio Group(1).....	47
รูปที่ 4.8 ขั้นตอนการสร้างDynamic Action ในRadio Group(2).....	47
รูปที่ 4.9 ขั้นตอนการสร้างDynamic Action ในRadio Group(3).....	48
รูปที่ 4.10 ขั้นตอนการสร้างPage.....	49

รูป

หน้า

รูปที่ 4.11 ขั้นตอนการสร้าง Interactive Grid(1)	49
รูปที่ 4.12 ขั้นตอนการสร้าง Interactive Grid(2)	50
รูปที่ 4.13 ขั้นตอนการQueryของตารางกำหนด รหัส ชื่อ ผู้ใช้งาน	50
รูปที่ 4.14 ขั้นตอนการQueryของตารางข้อมูล	51
รูปที่ 4.15 ขั้นตอนการQueryของตารางงวดการประมวลผล.....	51
รูปที่ 4.16 ขั้นตอนการอนุมัติ Edit ข้อมูล(1)	52
รูปที่ 4.17 ขั้นตอนการอนุมัติ Edit ข้อมูล(2)	52
รูปที่ 4.18 ขั้นตอนการสร้างปุ่มเพิ่มข้อมูลและบันทึก	52
รูปที่ 4.19 การเพิ่มแถวโดยกำหนดJavaScript	53
รูปที่ 4.20 ขั้นตอนการใส่ Static Id(1)	53
รูปที่ 4.21 ขั้นตอนการใส่ Static Id(2)	54
รูปที่ 4.22 ขั้นตอนกำหนด Master Detail.....	54
รูปที่ 4.23 ขั้นตอนผูกปุ่มให้กับProcess(1)	54
รูปที่ 4.24 ขั้นตอนผูกปุ่มให้กับProcess(2)	55
รูปที่ 4.25 UI ของหน้ากำหนดชื่อ รหัส ผู้ใช้งาน	55
รูปที่ 4.26 ขั้นตอนการสร้าง Static Content(1)	56
รูปที่ 4.27 ขั้นตอนการสร้าง Static Content(2)	56
รูปที่ 4.28 ขั้นตอนการสร้างInteractive Grid Match เงินได้/เงินหัก(1)	57
รูปที่ 4.29 ขั้นตอนการสร้างInteractive Grid Match เงินได้/เงินหัก(2)	57
รูปที่ 4.30 ขั้นตอนEnable Interactive Grid Match เงินได้/เงินหัก(1)	58
รูปที่ 4.31 ขั้นตอนEnable Interactive Grid Match เงินได้/เงินหัก(2)	58
รูปที่ 4.32 ขั้นตอนการสร้างปุ่มสำหรับบันทึกข้อมูลMatch เงินได้/เงินหัก	59
รูปที่ 4.33 ขั้นตอนการสร้างProcess Match เงินได้/เงินหัก.....	60
รูปที่ 4.34 ขั้นตอนลิ้งค์ไปยังหน้าที่ต้องการ(1).....	60
รูปที่ 4.35 ขั้นตอนลิ้งค์ไปยังหน้าที่ต้องการ(2).....	60
รูปที่ 4.36 ขั้นตอนลิ้งค์ไปยังหน้าที่ต้องการ(3).....	61

รูป

หน้า

รูปที่ 4.37 UI Match หน้าเงินได้/เงินหัก.....	61
รูปที่ 4.38 ขั้นตอนการสร้างInteractive Grid กำหนดระดับงวดประมวลผล	62
รูปที่ 4.39 ขั้นตอนการสร้างInteractive Grid งวดการประมวล	63
รูปที่ 4.40 ขั้นตอนการสร้าง Dynamic Actionปุ่มเพิ่มข้อมูลตารางระดับ(1)	63
รูปที่ 4.41 ขั้นตอนการสร้าง Dynamic Actionปุ่มเพิ่มข้อมูลตารางระดับ(2)	64
รูปที่ 4.42 ขั้นตอนการผูกปุ่มButtonกับProcessงวดการประมวล(1).....	64
รูปที่ 4.43 ขั้นตอนการผูกปุ่มButtonกับProcessงวดการประมวล(2).....	64
รูปที่ 4.44 ขั้นตอนการกำหนด Master Detail งวดการประมวล.....	65
รูปที่ 4.45 UI ของกำหนดระดับงวดการประมวล.....	65
รูปที่ 4.46 ขั้นตอนการสร้าง Interactive Grid ลดหย่อนภาษี (Tax Allowance Fix).....	66
รูปที่ 4.47 ขั้นตอนการสร้าง Interactive Grid ค่าสะสม (YTOD).....	67
รูปที่ 4.48 ปุ่มเพิ่มข้อมูลและบันทึกข้อมูล	67
รูปที่ 4.49 ขั้นตอนการ Enable เพิ่มเพิ่มแถวข้อมูล(1).....	68
รูปที่ 4.50 ขั้นตอนการ Enable เพิ่มเพิ่มแถวข้อมูล(2).....	68
รูปที่ 4.51 ขั้นตอนการสร้างไอเทมโชว์Flag.....	70
รูปที่ 4.52 ขั้นตอนการสร้างDynamic Action Disable Button(1).....	71
รูปที่ 4.53 ขั้นตอนการสร้างDynamic Action Disable Button(2).....	71
รูปที่ 4.54 ขั้นตอนการสร้างDynamic Action Disable Button(3).....	72
รูปที่ 4.55 ขั้นตอนการสร้างDynamic Action Show flag(1).....	72
รูปที่ 4.56 ขั้นตอนการสร้างDynamic Action Show flag(2).....	73
รูปที่ 4.57 UI ของหน้าบันทึกเงินล่วงเวลา	74
รูปที่ 4.58 ขั้นตอนการสร้างInteractive Grid กำหนดประเภทการลา	75
รูปที่ 4.59 ขั้นตอนการแบบฟอร์มเพิ่มข้อมูลประเภทการลา	75
รูปที่ 4.60 ขั้นตอนการแบบฟอร์มเพิ่มข้อมูลประเภทการลา	75
รูปที่ 4.61 ขั้นตอนการเขียนplsql insert , update , delete.....	78
รูปที่ 4.62 ขั้นตอนการสร้างปุ่มเพิ่มข้อมูลกำหนดประเภทการลา	78

รูป

หน้า

รูปที่ 4.63 ขั้นตอนการกำหนดBranch(1)	78
รูปที่ 4.64 ขั้นตอนการกำหนดBranch(2)	78
รูปที่ 4.65 ขั้นตอนการกำหนดBranch(3)	79
รูปที่ 4.66 ขั้นตอนการสร้างInteractive Gridตาราง โครงสร้างกำหนดผู้อนุมัติ(1).....	80
รูปที่ 4.67 ขั้นตอนการสร้างInteractive Gridตารางกำหนดผู้อนุมัติส่วนรายละเอียด(1).....	81
รูปที่ 4.68 ขั้นตอนการสร้างInteractive Gridตารางกำหนดผู้อนุมัติส่วนรายละเอียด(2).....	81
รูปที่ 4.69 ขั้นตอนการสร้างปุ่มเพิ่มข้อมูลและบันทึกข้อมูล.....	82
รูปที่ 4.70 ขั้นตอนการสร้างDynamic Action ปุ่มเพิ่มข้อมูลที่ตาราง โครงสร้างกำหนดผู้อนุมัติ(1).....	82
รูปที่ 4.71 ขั้นตอนการสร้างDynamic Actionปุ่มเพิ่มข้อมูลที่ตาราง โครงสร้างกำหนดผู้อนุมัติ(2).....	83
รูปที่ 4.72 ขั้นตอนการสร้างDynamic Actionปุ่มเพิ่มข้อมูลที่กำหนดผู้อนุมัติส่วนรายละเอียด(1)	83
รูปที่ 4.73 ขั้นตอนการสร้างDynamic Actionปุ่มเพิ่มข้อมูลที่กำหนดผู้อนุมัติส่วนรายละเอียด(2)	84
รูปที่ 4.74 ขั้นตอนการปรับคอลัมน์เป็นSelect list(1).....	84
รูปที่ 4.75 ขั้นตอนการปรับคอลัมน์เป็นSelect list(2).....	85
รูปที่ 4.76 UI หน้าโครงสร้างกำหนดผู้อนุมัติ.....	86
รูปที่ 4.77 รูปแบบของจำนวนเงินก่อนทำ Dynamic Format.....	86
รูปที่ 4.78 ขั้นตอนการสร้างDynamic Action Set format(1)	87
รูปที่ 4.79 ขั้นตอนการสร้างDynamic Action Set format(2)	87
รูปที่ 4.80 ขั้นตอนการสร้างDynamic Action Set format(3)	88
รูปที่ 4.81 รูปแบบของจำนวนเงินหลังทำ Dynamic Format	88
รูปที่ 4.82 ไอเทมรับค่าPKพนักงานและชื่อ	89
รูปที่ 4.83 ขั้นตอนการสร้าง Dynamic Action Set Select list(1)	89
รูปที่ 4.84 ขั้นตอนการสร้าง Dynamic Action Set Select list(2)	90
รูปที่ 4.85 ขั้นตอนการสร้างProcessส่งค่า(1).....	90
รูปที่ 4.86 ขั้นตอนการสร้างProcessส่งค่า(2).....	92
รูปที่ 4.87 ขั้นตอนการสร้างSelect list สำหรับเลือกกลุ่มPayroll(1).....	92
รูปที่ 4.88 ขั้นตอนการสร้างSelect list สำหรับเลือกกลุ่มPayroll(2).....	93

รูป

หน้า

รูปที่ 4.89 ขั้นตอนการสร้างปุ่มส่งค่าRequest	93
รูปที่ 4.90 ขั้นตอนการสร้างProcessรับค่าRequest	94
รูปที่ 4.91 หน้าข้อมูลพนักงานที่มีเลือกกลุ่ม Payroll.....	94
รูปที่ 4.92 ขั้นตอนดาวน์โหลดPlug-in	95
รูปที่ 4.93 ขั้นตอนลงPlug-in Google Organization Chart (1).....	95
รูปที่ 4.94 ขั้นตอนลงPlug-in Google Organization Chart (2).....	95
รูปที่ 4.95 ขั้นตอนลงPlug-in Google Organization Chart (3).....	96
รูปที่ 4.96 ขั้นตอนลงPlug-in Google Organization Chart (4).....	96
รูปที่ 4.97 ขั้นตอนลงPlug-in Google Organization Chart(5).....	96
รูปที่ 4.98 ขั้นตอนเรียกใช้Plug-in Google Organization Chart(1)	97
รูปที่ 4.99 ขั้นตอนเรียกใช้Plug-in Google Organization Chart(2)	97
รูปที่ 4.100 ขั้นตอนเรียกใช้Plug-in Google Organization Chart(3)	98
รูปที่ 4.101 Google Organization Chart	98
รูปที่ 4.102 ขั้นตอนการปรับรูปแบบวันที่	99
รูปที่ 4.103 ขั้นตอนการสร้างDynamic Action Refresh(1).....	102
รูปที่ 4.104 ขั้นตอนการสร้างDynamic Action Refresh(2).....	102
รูปที่ 4.105 ขั้นตอนการสร้างDynamic Action Refresh(3).....	103
รูปที่ 4.106 ขั้นตอนการสร้างDynamic Action Refresh(4).....	104
รูปที่ 4.107 ขั้นตอนการลง Plug-in Tooltip (1)	105
รูปที่ 4.108 ขั้นตอนการลง Plug-in Tooltip (2)	105
รูปที่ 4.109 ขั้นตอนการลง Plug-in Tooltip (3)	105
รูปที่ 4.110 ขั้นตอนการลง Plug-in Tooltip (4)	106
รูปที่ 4.111 ขั้นตอนการเรียกใช้ Tooltip (1)	106
รูปที่ 4.112 ขั้นตอนการเรียกใช้ Tooltip (2)	107
รูปที่ 4.113 ขั้นตอนการเรียกใช้ Tooltip (3)	107
รูปที่ 4.114 ขั้นตอนการเรียกใช้ Tooltip (4)	108

รูป

หน้า

รูปที่ 4.115 ขั้นตอนการเรียกใช้ Tooltip (5)	108
รูปที่ 4.116 ขั้นตอนการเรียกใช้ Tooltip (6)	109
รูปที่ 4.117 ตัวอย่างTooltip	109
รูปที่ 4.118 ข้อมูลพนักงานของกลุ่มStaff.....	110
รูปที่ 4.119 เงื่อนไขเงินได้, สิทธิประโยชน์.....	111
รูปที่ 4.120 กำหนดงวดการประมวลผล	111
รูปที่ 4.121 ประมวลเงินเดือนและภาษี.....	112
รูปที่ 4.122 ปิดงวดเงินเดือน.....	112
รูปที่ 4.123 ข้อมูลเงินเดือน.....	113



TNi

สารบัญตาราง

ตาราง

หน้า

ตารางที่ 1.4 ตารางหน้าที่ที่ได้รับมอบหมาย.....	7
ตารางที่ 3.1 แผนปฏิบัติงานโครงการ	38

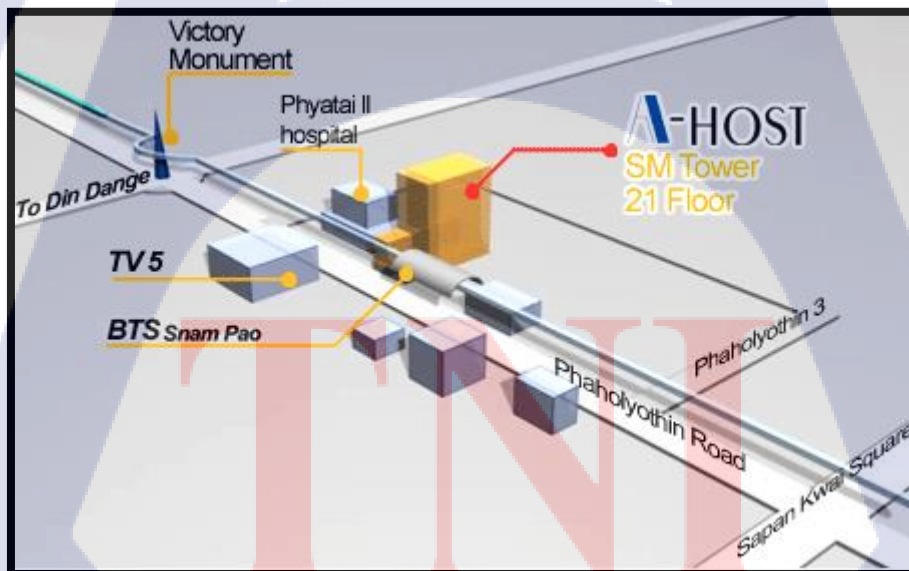


บทที่ 1

บทนำ

1.1 ชื่อและที่ตั้งของสถานประกอบการ

ชื่อสถานประกอบการ : บริษัท เอ-โฮสต์ จำกัด
A-HOST Company Limited
ที่ตั้งสถานประกอบการ : 979/53-55 ชั้น 21 อาคาร SM Tower ถนน มหลโยธิน
แขวงสามเสนใน เขตพญาไท
กรุงเทพฯ 10400
โทรศัพท์ : (66) 2298-0625-32
โทรสาร : (66) 2298-0053



รูปที่ 1.1 แผนที่ บริษัท เอ-โฮสต์ จำกัด

1.2 ลักษณะธุรกิจของสถานประกอบการ หรือการให้บริการหลักขององค์กร

บริษัทเอ-โฮสต์จำกัดได้ก่อตั้งขึ้นเมื่อปีพ.ศ.2542 ในฐานะหนึ่งบริษัทในเครือของบริษัทซิสเต็มส์คอร์ปอเรชั่น (มหาชน) จำกัดและเป็นผู้เชี่ยวชาญด้านการบริการจัดวางระบบสารสนเทศ (Information Technology : IT) และบริการเสริมต่างๆสำหรับลูกค้าตั้งแต่ธุรกิจขนาดย่อมไปจนถึงขนาดกลางธุรกิจหลักของบริษัทเอ-โฮสต์คือการให้บริการโฮสติ้ง (Hosting) ซึ่งเป็นการเปิดเครื่องคอมพิวเตอร์แม่ข่ายในการให้เช่าพื้นที่เพื่อวางระบบและบริการระบบสารสนเทศด้วยผลิตภัณฑ์ของออราเคิล (Oracle) เป็นซอฟต์แวร์สำหรับการวางแผนบริหารทรัพยากรของระดับแนวหน้าของโลก

เอ-โฮสต์ถือกำเนิดขึ้นจากกลุ่มผู้เชี่ยวชาญด้านสารสนเทศท่ามกลางภาวะเศรษฐกิจตกต่ำทั่วภูมิภาคเอเชีย แต่เอ-โฮสต์ก็สามารถเติบโตอย่างรวดเร็วและมั่นคงตั้งแต่แรกก่อตั้งด้วยจุดแข็งในฐานะผู้บุกเบิกธุรกิจโฮสติ้ง เซอร์วิสพร้อมทั้งนำธุรกิจแนวใหม่อย่างการให้บริการระบบโปรแกรมประยุกต์หรือ ASP (Application Services Provider) เข้ามาให้บริการเป็นรายแรกในเมืองไทย อีกทั้งยังถือเป็นผู้ให้บริการรายแรกนอกประเทศสหรัฐอเมริกาด้วย

ในฐานะผู้นำในอุตสาหกรรมนี้เป็นเวลามากกว่า 10 ปีเอ-โฮสต์ได้เสริมสร้างความแข็งแกร่งทางธุรกิจด้วยบริการที่มีความโดดเด่นและรวบรวมเอาทรัพยากรบุคคลซึ่งได้สั่งสมประสบการณ์และความชำนาญไว้อย่างพร้อมเพรียงส่งผลให้ศูนย์ข้อมูลของเอ-โฮสต์ในปัจจุบันมีความสมบูรณ์ด้วยกลุ่มเซิร์ฟเวอร์ (Server) ที่เชื่อมต่อกันในลักษณะการจัดกลุ่ม (Clustering) ซึ่งเปี่ยมสมรรถนะสามารถให้บริการแก่ผู้ใช้งานจำนวนมากได้ในเวลาเดียวกัน

นอกจากนี้เอ-โฮสต์ยังติดตั้งระบบรักษาความปลอดภัยระบบสำรองข้อมูลและระบบบริหารจัดการ รวมถึงอุปกรณ์ต่างๆอย่างครบครันเพื่อให้เอ-โฮสต์สามารถตอบสนองต่อระดับความต้องการในระดับสูงสุดที่ถูกคาดหวังได้ตลอดจนเป็นการสร้างความมั่นใจให้แก่ลูกค้าที่ใช้บริการโฮสติ้งและแอปพลิเคชันต่างๆของเอ-โฮสต์ว่าจะได้รับทั้งประสิทธิภาพและความปลอดภัยอย่างครบครันธุรกิจการให้บริการระบบโปรแกรมประยุกต์ในรูปแบบ ASP เอ-โฮสต์ไม่เพียงแต่ให้บริการด้านโปรแกรมประยุกต์ด้านการดำเนินธุรกิจทางอิเล็กทรอนิกส์ระดับโลกของออราเคิลพร้อมโครงสร้างพื้นฐานทางเทคโนโลยีสารสนเทศเท่านั้น แต่ยังมีบริการที่ครอบคลุมตั้งแต่การให้คำปรึกษาการสนับสนุนและการให้บริการทั่วไปอย่างพร้อมพร้อมครบครัน

นอกจากธุรกิจโฮสติ้งและธุรกิจการให้บริการโปรแกรมประยุกต์ในรูปแบบ ASP ซึ่งถือเป็นธุรกิจหลัก เอ-โฮสต์ยังดำเนินหน้าธุรกิจอย่างต่อเนื่องโดยการขยายหน่วยงานใหม่เพิ่มขึ้นนั่นก็คือ Core Technology Division หน่วยงานเทคโนโลยีหลักที่ให้คำตอบเปิดเสรีแก่ลูกค้าด้วยระบบฐานข้อมูลและเครื่องมือต่างๆ ของออราเคิลซึ่งช่วยเพิ่มความสามารถของลูกค้าในการออกแบบพัฒนาปรับเปลี่ยนระบบแอปพลิเคชันให้เหมาะสมกับธุรกิจนั้นๆ ภายใต้คำปรึกษาแนะนำและการวางระบบของเอ-โฮสต์ลูกค้าสามารถบริหารระบบฐานข้อมูลของตนเองและดูแลระบบดังกล่าวได้อย่างมีประสิทธิภาพ

ตลอดระยะเวลามากกว่า 10 ปีในการดำเนินธุรกิจของเอ-โฮสต์ไม่เพียงแต่ในฐานะผู้บุกเบิกธุรกิจโฮสติ้งและธุรกิจการให้บริการ โปรแกรมประยุกต์ในรูปแบบ ASP เท่านั้นแต่เอ-โฮสต์ยังได้ทำการติดตั้งระบบสารสนเทศรวมทั้งผลิตภัณฑ์ของออราเคิลให้กับลูกค้าจนประสบความสำเร็จเป็นจำนวนมากซึ่งหลายรายเป็นหนึ่งในร้อยบริษัทชั้นนำของประเทศไทยแต่สิ่งที่สำคัญกว่านั้นก็คือการที่เอ-โฮสต์ได้สานสัมพันธ์กับลูกค้าและพันธมิตรทางธุรกิจอย่างแนบแน่นจนกลายเป็นหุ้นส่วนทางกลยุทธ์และได้รับแต่งตั้งให้เป็น OCAP (Oracle Certified Advantage Partner) รายแรกในประเทศไทย



รูปที่ 1.2 รางวัลที่บริษัท A-HOST ได้รับ

ปัจจุบันเอ-โฮสต์มีประเภทของสินค้าและบริการซึ่งสามารถแบ่งออกเป็น 3 กลุ่มใหญ่ได้แก่

1.2.1 Hosting & Outsource Service

เอ-โฮสต์ได้ปรับปรุงและขยายการบริการ Hosting และการให้บริการภายนอก (Outsource) จนสามารถครอบคลุมความต้องการของลูกค้าได้อย่างหลากหลายโดยยึดหลักในการให้บริการที่เรียกว่า “Peace of Mind for the customer” ซึ่งหมายถึงการที่จะทำงานให้กับลูกค้าแบบครบวงจรเพื่อให้ลูกค้าจะสามารถใช้งานระบบสารสนเทศได้อย่างมีประสิทธิภาพได้อย่างสบายใจไร้ความกังวลต่อความเสี่ยงต่างๆไม่ว่าจะเป็นเรื่องของปัญหาทางเทคนิคการจัดทำระบบและข้อมูลสำรองการปรับแต่งระบบให้ได้ประสิทธิภาพสูงสุด (Performance Tuning) และที่สำคัญที่สุดคือการที่เข้ามารับภาระในด้านการบริหารจัดการบุคลากรทางด้านสารสนเทศทั้งหมดแทนลูกค้า

การใช้บริการ Hosting และ Outsource จะทำให้ลูกค้าสามารถทุ่มเทกำลังสมองเวลาและทรัพยากรขององค์กรให้กับธุรกิจที่เป็นแกนหลัก (Core Business) ซึ่งเป็นสิ่งที่ลูกค้าถนัดกว่าโดยทั่วไปแล้วบริการ Hosting และ Outsource จะประกอบด้วยส่วนประกอบและบริการย่อยๆดังต่อไปนี้

1. High Availability and High Performance IT Infrastructure
2. Dedicated or Co-Location Service
3. Disaster Site
4. Oracle E-Business Application (ERP, CRM, SCM)
5. ERP Implementation Service
6. System and Database Administration
7. Help Desk
8. On-Request Service i.e. On-Site Support, Software Customization

ทั้งนี้การบริการ Hosting และ Outsource สามารถครอบคลุมได้ทั้งระบบที่ใช้เทคโนโลยีของออรากิลและระบบที่ใช้เทคโนโลยีอื่นๆ

1.2.2 Oracle Core Technology Products and Advanced Services

เอ-โฮสต์เป็นผู้นำในการดำเนินธุรกิจในฐานะผู้แทนจำหน่ายเพิ่มมูลค่าให้กับออรากิลโดยไม่เพียงแต่ทำหน้าที่ในการจัดจำหน่ายสินค้าในกลุ่มแกนหลักของเทคโนโลยี (Core Technology) ของออรากิลทุกประเภทแต่ยังมีทีมผู้เชี่ยวชาญที่จะให้การสนับสนุนและบริการเสริมอย่างครบวงจรแก่บริษัทลูกค้าและ

ลูกค้าไม่ว่าจะเป็นการร่วมจัดกิจกรรมทางการตลาดการฝึกอบรมการติดตั้งระบบและการให้คำปรึกษาเพื่อแก้ไขปัญหาต่างๆสินค้าและบริการที่อยู่ในกลุ่ม Oracle Core Technology Products and Advanced Services ได้แก่

1. Oracle Database and database options
2. Oracle Business Intelligence Suite
3. Business Partner Development
4. System Installation, Integration and Optimization
5. Oracle Fusion Middleware (รวมถึง BEA)
6. สินค้าอื่นๆทุกประเภทของออราเคิล
7. Marketing and Lead Generation Activities
8. SOA-Based Development and Implementation

ผลสำเร็จในการดำเนินธุรกิจประเภทนี้ทั้งในด้านการตลาดและบริการทำให้เอ-โฮสต์ได้รับรางวัล ASEAN Partner of the Year ในปี 2005

1.2.3 Oracle Enterprise Performance Management (EPM) และ Hyperion Business Intelligence Products and Services

ความต้องการสูงสุดประการหนึ่งของผู้บริหารในการนำเอาระบบสารสนเทศมาใช้ในองค์กรไม่ว่าจะเป็นภาครัฐหรือเอกชนคือการทำให้ผู้บริหารสามารถได้ข้อมูลที่แสดงให้เห็นถึงสถานะในการดำเนินธุรกิจได้อย่างแม่นยำรวดเร็วและการนำเอาข้อมูลมาวิเคราะห์และวางแผนทั้งในระดับปฏิบัติการและในระดับกลยุทธ์เพื่อให้ธุรกิจสามารถได้เปรียบในการแข่งขันปรับตัวตามเศรษฐกิจได้ในทุกสถานการณ์ Oracle Enterprise Performance Management (EPM) และ Hyperion Business Intelligence จัดเป็นซอฟต์แวร์ชั้นนำของโลกที่สามารถสนองต่อความต้องการในลักษณะดังกล่าวได้เป็นอย่างดี

เอ-โฮสต์มีทีมงานที่ปรึกษาที่มีประสบการณ์ทั้งทางด้านธุรกิจและทางด้านเทคนิครวมถึงความเข้าใจในระบบ ERP ของออราเคิลอย่างลึกซึ้งจึงทำให้สามารถให้บริการที่ปรึกษาเพื่อออกแบบติดตั้งเชื่อมโยงและปรับใช้ระบบให้กับลูกค้าจนเกิดประสิทธิภาพสูงสุด อีกทั้งยังมีความยืดหยุ่นและให้การตอบสนองที่เร็วกว่าเมื่อเทียบกับการว่าจ้างที่ปรึกษาจากต่างประเทศ

1.3 รูปแบบการจัดองค์กรและการบริหารองค์กร



รูปที่ 1.3 คณะผู้บริหาร A-HOST Company Limited

1.4 ตำแหน่งและหน้าที่งานที่นักศึกษาได้รับมอบหมาย

ตำแหน่ง : Programmer

หน่วยงาน : BS

ตารางที่ 1.4 ตารางหน้าที่ที่ได้รับมอบหมาย

KPI Description	Expected Results
1. พัฒนาระบบ Payroll ด้วย Apex5	- สามารถพัฒนาระบบได้อย่างมีประสิทธิภาพ - โปรแกรมผ่านการทดสอบบนเครื่อง UAT (ทำเอกสาร Check List) - ทำเอกสารสำหรับการพัฒนาระบบ เช่น Code Diff. หรือ Programming Flow - สามารถช่วยกำหนด Test Case ได้ - จัดทำเอกสาร User Manual - จัดทำเอกสาร Programmer Manual
2. ศึกษา Technology ใหม่ หรือหัวข้อที่น่าสนใจ เพื่อสนับสนุนงานโครงการต่างๆที่ได้รับมอบหมาย	- สามารถนำไปใช้งานได้จริง - สามารถถ่ายทอดให้ผู้อื่นได้ - ศึกษาเทคโนโลยี และจัดทำ Programmer Manual 1. ศึกษาและ POC ในส่วนของเทคโนโลยีที่จะพัฒนาในส่วนของการพัฒนาระบบ Payroll
3. จัดทำ IWL	- นำเสนอ IWL เป็นประจำทุกเดือน

1.5 พนักงานที่ปรึกษา และ ตำแหน่งของพนักงานที่ปรึกษา

นางสาวกัทรทา หมีนเมือง

ตำแหน่ง : Application Support

Email : Patta@a-host.co.th

เบอร์โทรศัพท์ : 0824258155

1.6 ระยะเวลาที่ปฏิบัติงาน

ปฏิบัติงานสหกิจศึกษาเป็นเวลา 4 เดือน วันที่ 4 มิถุนายน 2561 ถึง วันที่ 28 กันยายน 2561 ก่อนสหกิจมี
การอบรมเป็นเวลา 2 เดือน

1.7 ที่มาและความสำคัญของปัญหา

ปัจจุบันองค์กรทางธุรกิจส่วนใหญ่ต้องการระบบที่สามารถช่วยบริหารจัดการเงินเดือน ลดความ
ยุ่งยากในการจ่ายเงิน สามารถบรรลุความต้องการในการลดเวลาการทำงานขององค์กรในการเก็บข้อมูล
ซึ่งประเด็นที่สำคัญที่น่าจับตาในการบริหารระบบเงินเดือน คือ การสร้างและพัฒนาระบบเงินเดือนให้
พร้อมบริการอย่างมีความน่าเชื่อถือสูง(Reliability) เนื่องจากหากเป็นองค์กรเชิงธุรกิจหรือเชิงต่างๆ
แน่นอนว่าหากเกิดปัญหาข้อผิดพลาดในการเก็บข้อมูลพนักงาน ความล่าช้า ความผิดพลาดในการ
คำนวณ ไม่ว่าจะเป็น เงินล่วงเวลา เงินเดือน หรือ รายการเงินได้ เงินหัก ย่อมเกิดความเสียหายภายใน
องค์กรอย่างแน่นอน

ดังนั้นการพัฒนาระบบเงินเดือนให้สามารถคำนวณได้อย่างถูกต้อง รวดเร็ว ใช้งานง่าย ลดเวลาใน
การทำงานของ HR ในการเก็บข้อมูล สามารถแบ่งแยกตามแผนก แผนงาน ฝ่ายได้ จึงนำเทคโนโลยี
Oracle Apex มาช่วยในการจัดการบริหารงาน

1.8 วัตถุประสงค์หรือจุดมุ่งหมายของโครงการ

การศึกษาในหัวข้อ Development and Support Payroll system มีวัตถุประสงค์เพื่อช่วยจัดการงาน
บัญชีเกี่ยวกับระบบเงินเดือนและค่าแรง สามารถคำนวณรายการเงินได้ เงินหักในกรณีต่างๆ คัดคำนวณเงิน
ล่วงเวลา ประกันสังคม ฯลฯ รวมถึงยืดหยุ่นในการทำงานในการปรับเปลี่ยนค่ากำหนดต่างๆ

1.9 ผลที่คาดว่าจะได้รับจากการปฏิบัติงานหรือโครงการที่ได้รับมอบหมาย

1. นักศึกษาสามารถทำงานภายใต้สถานการณ์ทำงานจริงได้
2. นักศึกษาสามารถนำความรู้ที่ได้จากการปฏิบัติงานสหกิจไปใช้ประกอบอาชีพ
3. นักศึกษาสามารถทำงานร่วมกับผู้อื่นได้เป็นอย่างดี
4. มีความรับผิดชอบในหน้าที่การงานที่ได้รับมอบหมาย
5. ระบบสามารถบริหารเงินเดือนได้อย่างมีประสิทธิภาพ
6. ระบบสามารถนำไปใช้งานได้จริง
7. ระบบสามารถคำนวณรายการต่างๆได้

1.10 นิยามศัพท์เฉพาะ

Transaction	: กลุ่มคำสั่งที่ใช้จัดการข้อมูลเพื่อการประมวลผล
Time Attendance	: บันทึกเวลาการทำงาน
Personnel Module	: ระบบพนักงาน
OT	: ทำงานล่วงเวลา

บทที่ 2

ทฤษฎีและเทคโนโลยีที่ใช้ในการปฏิบัติงาน

ในการปฏิบัติงานสหกิจศึกษาครั้งนี้ เป็นการนำความรู้ทางด้านทฤษฎีและเทคโนโลยีมาใช้ในการปฏิบัติงานทุกส่วนตลอดการปฏิบัติงานสหกิจศึกษา ซึ่งเป็นการนำความรู้ทั้งที่เคยเรียนมาประยุกต์ใช้ และเป็นการศึกษาเรียนรู้สิ่งใหม่ ๆ ที่ได้จากการปฏิบัติงาน

2.1 ภาษา SQL (Structured Query Language)

ภาษา SQL ย่อมาจาก Structured Query Language หรือภาษาสอบถามข้อมูลเป็นภาษาทางด้านฐานข้อมูลที่สามารถสร้างและปฏิบัติการกับฐานข้อมูลแบบสัมพันธ์ (Relational Database) โดยเฉพาะและเป็นภาษาที่มีลักษณะคล้ายกับภาษาอังกฤษ ภาษา SQL ถูกพัฒนาขึ้นจากแนวคิดของ Relational Calculus และ Relation Algebra เป็นหลักภาษา SQL เริ่มพัฒนาครั้งแรกโดยหน่วยงาน Almaden Research Center ของบริษัท IBM โดยมีชื่อเริ่มแรกว่า “ซีเควล” (Sequel) ต่อมาได้เปลี่ยนชื่อเป็น “เอสคิวแอล” (SQL) หลังจากนั้น ภาษา SQL ได้ถูกนำมาพัฒนาโดยผู้ผลิตซอฟต์แวร์ด้านระบบจัดการฐานข้อมูลเชิงสัมพันธ์จนเป็นที่นิยมกันอย่างแพร่หลายในปัจจุบันโดยผู้ผลิตแต่ละรายก็พยายามที่จะพัฒนาระบบจัดการฐานข้อมูลของตนให้มีลักษณะเด่นเฉพาะขึ้นมาทำให้รูปแบบการใช้คำสั่ง SQL มีรูปแบบที่แตกต่างกันไปบ้างเช่น Oracle Access SQL Base ของ Sybase Ingres หรือ SQL Server ของ Microsoft เป็นต้น ดังนั้นในปีค.ศ. 1986 ทางด้าน American National Standards Institute (ANSI) จึงได้กำหนดมาตรฐานของ SQL ขึ้นอย่างใดก็ตามโปรแกรมฐานข้อมูลที่ขายในท้องตลาดได้ขาย SQL ออกไปจนเกินข้อกำหนดของ ANSI โดยเพิ่มคุณสมบัติอื่นๆที่คิดว่าประโยชน์เข้าไปอีกแต่โดยหลักทั่วไปก็ยังปฏิบัติตามมาตรฐานของ ANSI

2.1.1 ประเภทของคำสั่งของภาษา SQL

ภาษา SQL เป็นภาษาที่ใช้งานได้ตั้งแต่ระดับเครื่องคอมพิวเตอร์ส่วนบุคคลไปจนถึงระดับเมนเฟรมประเภทของคำสั่งในภาษา SQL (The subdivision of SQL) แบ่งออกเป็น 3 ประเภท

2.1.1.1 คำสั่งการสร้าง (Create)

คำสั่งการสร้างตารางและอินเด็กซ์จะกำหนดชื่อตารางและกำหนดลักษณะข้อมูลเป็นคอลัมน์ต่างๆที่สร้างขึ้นในตารางรวมไปถึงชนิดของข้อมูลของแต่ละคอลัมน์นั้นในโครงสร้างของคำสั่งการสร้างตารางมีรูปแบบไวยากรณ์ดังต่อไปนี้

```
CREATE TABLE <table name>
(<column name>< data type> [<size>]
[[constraint <Constraint name>] constraint type]
[, <column name><data type> [<size>],.....]);
```

CREATE TABLE	คำสั่งที่ต้องมีทุกครั้งที่ต้องการสร้างตาราง
Table name	ชื่อตารางที่ต้องการสร้าง
Column name	ชื่อของคอลัมน์แต่ละคอลัมน์
Data type	ชนิดข้อมูลของคอลัมน์นั้นๆ
Constraint	ข้อกำหนดของคอลัมน์
Constraint name	ชื่อของข้อกำหนดที่ต้องการสร้างให้กับคอลัมน์
Constraint type	ประเภทของข้อกำหนด

2.1.1.1.1 การสร้างตารางสามารถกำหนดข้อจำกัด (Constraint)

การกำหนดข้อจำกัด (Constraint) ลงในค่าต่างๆที่จะป้อนลงในคอลัมน์ต่างๆของตารางได้ การกำหนดข้อจำกัดเป็นการควบคุมความถูกต้องสมบูรณ์ (Integrity) ที่จัดเก็บในฐานข้อมูลให้มีความถูกต้องตามที่ถูกกำหนดไว้หรือตามที่ควรจะเป็นการกำหนดข้อจำกัดทำให้ข้อมูลมีความเชื่อถือได้ การกำหนดข้อจำกัดจะทำให้ข้อมูลในตารางไม่สามารถรับค่าใดๆที่ไม่ตรงกับข้อจำกัดที่กำหนดไว้ การกำหนดข้อจำกัดที่เป็นการควบคุมความถูกต้องสมบูรณ์ (Integrity) ได้ดังนี้

- 1) NOT NULL การกำหนดไม่ให้ค่าใดค่าหนึ่งเป็นค่าว่าง
- 2) UNIQUE การกำหนดไม่ให้มีค่าซ้ำกัน
- 3) PRIMARY KEY การกำหนดคีย์หลักสามารถทำได้ 2 วิธี
- กำหนดให้คอลัมน์เดียวเป็นคีย์หลัก

- กำหนดให้คอลัมน์มากกว่า 1 คอลัมน์เป็นคีย์หลัก
- 4) FOREIGN KEY การกำหนดคีย์นอกซึ่งมีการเชื่อมโยงกับอีกตารางหนึ่ง
- 5) CHECK การกำหนดการตรวจสอบ
- 6) DEFAULT การกำหนดค่ามาตรฐานให้กับคอลัมน์นั้นๆเมื่อเป็นค่าว่าง

2.1.1.1.2 การลบโครงสร้างข้อมูล

สามารถทำได้ด้วยคำสั่ง DROP TABLE ซึ่งมีรูปแบบทั่วไปดังนี้

DROP TABLE <table name> CASCADE CONSTRAINTS;

DROP TABLE	เป็นคำสั่งที่ต้องมีทุกครั้งที่ต้องการลบโครงสร้างตาราง
Table name	ชื่อตารางที่ต้องการลบ
CASCADE CONSTRAINTS	ระบบจัดการฐานข้อมูลจะทำการลบข้อจำกัดต่างๆ(Constraint) ที่มีการอ้างอิงตารางด้วยทั้งหมด

2.1.1.1.3 การเปลี่ยนแปลงโครงสร้างตารางที่มีการสร้างไว้

การเพิ่มหรือลบบางคอลัมน์ที่เป็นโครงสร้างหลักของตารางออกหรือต้องการเปลี่ยนแปลงประเภทข้อมูลของคอลัมน์ซึ่งในกรณีที่ตารางมีข้อมูลและกำหนดโครงสร้างไปแล้วการแก้ไขโครงสร้างข้อมูลอาจมีผลกระทบกับข้อมูลที่มีอยู่แต่ในภาษา SQL สามารถใช้คำสั่งในการแก้ไขโครงสร้างข้อมูลได้ด้วยคำสั่งการเปลี่ยนแปลงโครงสร้างตารางรูปแบบของของคำสั่ง ALTER TABLE มี 3 แบบคือ

- 1) ALTER TABLE ใช้เพิ่มคอลัมน์
- 2) ALTER TABLE ใช้เปลี่ยนชื่อคอลัมน์
3. ALTER TABLE ใช้ลบข้อมูลคอลัมน์

คำสั่ง ALTER TABLE เป็นคำสั่งที่ใช้ในการแก้ไขปรับปรุงโครงสร้างตารางเมื่อจำเป็นที่ต้องปรับปรุงจากโครงสร้างเดิมตามที่ได้กำหนดไว้ตั้งแต่สร้างตารางในครั้งแรกคำสั่ง ALTER TABLE มีรูปแบบไวยากรณ์ดังต่อไปนี้


```
ALTER TABLE<table name>
```

```
modify <column_name> data type [SIZE];
```

ALTER TABLE	คำสั่งที่ต้องมีทุกครั้งที่ต้องการเปลี่ยนแปลงโครงสร้างตาราง
Table name	ชื่อตารางที่จะเปลี่ยนแปลง
Modify	คำสั่งการเปลี่ยนแปลง
Column_name	ชื่อคอลัมน์
Data type [SIZE]	ชนิดข้อมูลและขนาดของข้อมูล

2.1.1.1.4 ภาษาสำหรับการจัดการข้อมูล (Data Manipulation Language: DML)

ถือเป็นสิ่งสำคัญในภาษา SQL สำหรับการจัดการข้อมูล (Data manipulation Language: DML) ซึ่งเป็นภาษาที่ใช้ในการบันทึกข้อมูลการปรับปรุงข้อมูลและการลบข้อมูลภาษาสำหรับการจัดการข้อมูลเป็นส่วนประกอบหนึ่งในภาษา SQL คำสั่งในภาษาสำหรับการจัดการข้อมูลจะเป็นคำสั่งที่ช่วยในการจัดการข้อมูลภายใน โครงสร้างตารางที่สร้างขึ้นตัวอย่างของคำสั่งในภาษาสำหรับการจัดการข้อมูลจะเป็นคำสั่งการปรับปรุงข้อมูล ได้แก่การเพิ่มข้อมูล (INSERT) การปรับปรุง (UPDATE) และการลบข้อมูล (DELETE)

คำสั่งทั้ง 3 นี้เมื่อดำเนินการในภาษา SQL จะไม่แสดงผลลัพธ์ออกมาทางหน้าจอแต่ผลของคำสั่งจะมีผลต่อข้อมูลผู้ใช้สามารถดูผลของการใช้คำสั่งในการเพิ่มข้อมูลการปรับปรุงและการลบข้อมูลโดยใช้คำสั่งการเรียกค้นข้อมูล (SELECT)

2.1.1.2 คำสั่งการเพิ่มข้อมูลในตาราง

จะใช้คำสั่ง INSERT จะมีอยู่ 2 รูปแบบคือการเพิ่มข้อมูลเข้าไปทีละแถวและการเพิ่มข้อมูลโดยการดึงกลุ่มข้อมูลด้วยคำสั่ง ค้นหาข้อมูล

2.1.1.2.1 คำสั่งการเพิ่มข้อมูลทีละแถว

โดยระบุข้อมูลที่จะ INSERT เข้าไปโดยตรงโดยมีรูปแบบไวยากรณ์ดังต่อไปนี้

```
INSERT INTO<table name> [(column 1, column 2,...)]
```

```
VALUE (<value1, value2 ...>);
```

INSERT INTO	คำสั่งที่ต้องมีทุกครั้งที่ต้องการเพิ่มข้อมูล
Table name	ชื่อตารางที่จะเพิ่มข้อมูล
Column 1, column 2,...	คอลัมน์ที่ต้องการเพิ่มข้อมูล
Value1, Value2	ค่าที่ต้องการเพิ่มลงคอลัมน์

2.1.1.2.2 คำสั่งการเพิ่มข้อมูลโดยการดึงกลุ่มข้อมูลด้วยคำสั่งค้นหาข้อมูล

ในภาษาSQL สามารถใช้คำสั่ง INSERT ในการนำค่าหรือหาค่าจากตารางหนึ่งใส่ไว้ในอีกตารางหนึ่งโดยได้ค่านั้นมาจากการสอบถามข้อมูล โดยมีรูปแบบไวยากรณ์ดังต่อไปนี้

```
INSERT INTO<table name> [(column 1, column 2,...)]
SELECT statement;
```

INSERT INTO	คำสั่งที่ต้องมีทุกครั้งที่ต้องการเพิ่มข้อมูล
Table name	ชื่อตารางที่จะเพิ่มข้อมูล
SELECT statement	ประโยคคำสั่ง SELECT ที่ต้องการข้อมูลอีกตารางหนึ่ง ค่าข้อมูลของแต่ละคอลัมน์ที่ต้องการเพิ่ม

2.1.1.2.3 คำสั่งแก้ไขแถวข้อมูล

เมื่อมีการป้อนข้อมูลเข้าไปเก็บไว้ในตาราง กรณีที่ต้องการปรับปรุงแก้ไขข้อมูลสามารถทำได้ด้วยภาษา SQL การแก้ไขแถวข้อมูลเป็นการปรับปรุงหรือแก้ไขค่าคอลัมน์ซึ่งในคำสั่งปรับปรุงข้อมูลอาจมีมากกว่า 1 คอลัมน์ในแถวทุกแถวที่มีเงื่อนไขสอดคล้องกับที่ระบุไว้หลังคำว่า WHERE โดยมีรูปแบบไวยากรณ์ดังต่อไปนี้

```
UPDATE <table name> SET <column 1>[, column 2...] = <expression |subquery>
[WHERE<condition>];
```

UPDATE	คำสั่งที่ต้องมีทุกครั้งที่ต้องการปรับปรุงข้อมูล
Table name	ชื่อตารางที่ต้องการปรับปรุง
SET <column >	ชื่อคอลัมน์ที่ต้องการปรับปรุง

Expression

ค่าข้อมูลที่ต้องการปรับปรุง

WHERE<condition>

เงื่อนไขในการปรับปรุง

2.1.1.2.4 คำสั่งการลบข้อมูล

คำสั่งที่ใช้ในการลบแถวข้อมูลทุกแถวที่มีเงื่อนไขสอดคล้องกับที่ระบุไว้หลัง WHERE คำสั่งการลบข้อมูลมีรูปแบบไวยากรณ์ดังต่อไปนี้

DELETE FROM <table name>

[WHERE<condition>];

DELETE FROM

Table name

WHERE<condition>

คำสั่งที่ต้องมีทุกครั้งที่ต้องการลบข้อมูล

ชื่อตารางที่ต้องการลบข้อมูล

เงื่อนไขในการลบข้อมูล

2.1.2 การเรียกค้นข้อมูลอย่างง่าย

การเรียกค้นข้อมูลเป็นการสอบถามข้อมูลหรือ “Query” โดยการนำข้อมูลจากฐานข้อมูลมาแสดงออกทางจอภาพการสอบถามข้อมูลนี้ในภาษา SQL ใช้คำสั่ง SELECT โดยการเรียกค้นข้อมูลจะเป็นไปตามเงื่อนไขที่ผู้ใช้ข้อมูลระบุ

2.1.2.1 การเรียกดูทุกคอลัมน์ในตาราง

การเรียกดูข้อมูลสามารถเรียกดูได้มากกว่า 1 คอลัมน์ขึ้นไปโดยถ้ามีมากกว่า 1 คอลัมน์แต่ละคอลัมน์จะต้องกันด้วยเครื่องหมายจุลภาค (,) และถ้าต้องการดูทุกคอลัมน์จะใช้เครื่องหมายดอกจัน (*) หลัง SELECT การใช้คำสั่ง SELECT จะใช้ควบคู่กับคำสั่ง FROM เสมอในการเลือกตาราง โดยคำสั่ง SELECT แบบง่ายมีรูปแบบไวยากรณ์ดังต่อไปนี้

SELECT *

FROM <table name>;

SELECT *

คำสั่งที่ต้องมีทุกครั้งที่ต้องการเรียกค้นข้อมูลทุกคอลัมน์

FROM

กำหนดว่าให้เรียกดูข้อมูลได้จากตารางใดบ้าง

Table name ชื่อตารางที่ต้องการเรียกค้นข้อมูล

2.1.2.2 การเรียกค้นข้อมูลเฉพาะคอลัมน์ใดๆในตารางและการเปลี่ยนลำดับคอลัมน์

การใช้คำสั่ง SELECT ในการเรียกค้นข้อมูลเฉพาะคอลัมน์ที่สนใจทำได้โดยใช้เฉพาะคอลัมน์ที่ต้องการดูในส่วนของการคำสั่ง SELECT มีรูปแบบไวยากรณ์ดังต่อไปนี้

SELECT<column 1, column 2,...>

FROM<table name>;

SELECT	คำสั่งที่ต้องมีทุกครั้งที่ต้องการเรียกค้นข้อมูล
Column 1, column 2,...	คอลัมน์ที่ต้องการเรียกค้น
FROM	การกำหนดเรียกดูข้อมูลได้จากตาราง
Table name	ชื่อตารางที่ต้องการเรียกค้นข้อมูล

2.1.2.3 การเรียกดูข้อมูลจากตารางเฉพาะข้อมูลที่ต้องการ (WHERE)

สืบค้นข้อมูลได้อย่างเจาะจงมากกว่าการสืบค้นข้อมูลทั่วไปมีรูปแบบไวยากรณ์ดังต่อไปนี้

SELECT <column 1, column 2,...>

FROM<table name>

[WHERE<condition>;]

SELECT	คำสั่งที่ต้องมีทุกครั้งที่ต้องการเรียกค้นข้อมูล
Column 1, Column 2...	คอลัมน์ที่ต้องการเรียกค้น
FROM	การกำหนดให้เรียกดูข้อมูลได้จากตารางใดบ้าง
Table name	ชื่อตารางที่ต้องการเรียกค้นข้อมูล
WHERE<condition>	ส่วนของคำสั่งที่บอกเงื่อนไขที่จะใช้ในการค้นหาข้อมูล

2.1.2.4 การเรียกดูข้อมูลแบบมีเงื่อนไข (SQL SELECT DISTINCT CLAUSE)

คำสั่งที่ใช้สำหรับการระบุเงื่อนไขการเลือกข้อมูลในตาราง (Table) โดยทำการเลือกข้อมูล
ที่ซ้ำกันมาเพียงแค่ Record เดียว รูปแบบไวยากรณ์ดังต่อไปนี้

```
SELECT DISTINCT <column1,column2,... >
```

```
FROM <Table name>;
```

SELECT	คำสั่งที่ต้องมีทุกครั้งที่ต้องการเรียกค้นข้อมูล
DISTINCT	คำสั่งที่ใช้สำหรับเลือกข้อมูลไม่ซ้ำกัน
Column 1,column 2,...	คอลัมน์ที่ต้องการเรียกค้น
FROM	การกำหนดเรียกดูข้อมูลได้จากตาราง
Table name	ชื่อตารางที่ต้องการเรียกค้นข้อมูล

2.1.3 ชนิดข้อมูลที่ใช้ในภาษา SQL

ภาษา SQL การบรรจุข้อมูลลงในคอลัมน์ต่างๆของตารางจะต้องกำหนดชนิดของข้อมูล (data type) ให้แต่ละคอลัมน์ชนิดของข้อมูลนี้จะแสดงชนิดของค่าที่อยู่ในคอลัมน์ค่าทุกค่าในคอลัมน์ที่กำหนดจะต้องเป็นชนิดเดียวกันเช่นในตารางลูกค้าคอลัมน์ที่เป็นรายชื่อลูกค้าจะต้องเป็นตัวอักษรในขณะที่คอลัมน์จำนวนเงินที่ลูกค้าซื้อสินค้าจะต้องเป็นตัวเลข

ชนิดของข้อมูลของแต่ละคอลัมน์จะขึ้นกับลักษณะของข้อมูลแต่ละคอลัมน์ซึ่งแบ่งชนิดข้อมูลพื้นฐานในภาษา SQL ได้ดังนี้

2.1.3.1 ตัวหนังสือ (Character)

2.1.3.1.1 ตัวหนังสือแบบความยาวคงที่ (Fixed-length Character)

Char (n) หรือ Character (n) แทนประเภทของข้อมูลที่เป็นตัวอักษรใดๆที่มีความยาวของข้อมูลคงที่โดยมีความยาว n ตัวประเภทนี้จะมีการจองเนื้อที่ตามความยาวที่คงที่ตามที่กำหนดไว้ ชนิดของข้อมูลประเภทนี้จะเก็บความยาวของข้อมูลได้มากที่สุดได้ 2000 ไบต์

2.1.3.1.2 ตัวหนังสือแบบความยาวไม่คงที่ (Variable-length Character)

Varchar2 (n) แทนประเภทของข้อมูลที่เป็นตัวอักขระใดๆที่มีความยาวของข้อมูลไม่คงที่ โดยมีความยาว n ตัวซึ่งจะมีการจองเนื้อที่ความยาวของข้อมูลชนิดของข้อมูลประเภทนี้จะเก็บความยาวของข้อมูลได้มากที่สุดได้ 4000 ไบต์

2.1.3.2 จำนวนเลข (Numeric)

ในภาษา SQL อาจใช้ number (p,s) แทนจำนวนเลขที่ไม่มีจุดทศนิยมและจำนวนเลขที่มีจุดทศนิยม โดย p คือจำนวนตัวเลขทั้งหมด (รวมทศนิยม) และ s คือจำนวนตัวเลขหลังทศนิยม

2.1.3.3 ข้อมูลวันที่

2.1.3.3.1 วันที่และเวลา (Date/Time)

เป็นชนิดวันที่หรือเวลาในภาษา SQL จะใช้ date แทนข้อมูลวันที่ซึ่งจะมีหลายรูปแบบให้เลือกใช้เช่น yyyy-mm-dd (2018-12-31) dd.mm.yyyy (31-12-2018) หรือ dd/mm/yyyy (31/12/2018)

2.1.4 ตัวดำเนินการ (Operator)

การเรียกค้นข้อมูลอย่างมีเงื่อนไขตามหลักของภาษา SQL จะอยู่หลังคำสั่ง WHERE ซึ่งสามารถเปรียบเทียบตามตัวดำเนินการในภาษา SQL อาจแบ่งตัวดำเนินการได้เป็น 4 กลุ่มคือ

2.1.4.1 ตัวดำเนินการทางคณิตศาสตร์ (Arithmetic Operators)

ได้แก่ Operators ที่เป็น plus (+) minus (-), divide (/), multiply (*), and modulo (%) นอกจากการใช้เครื่องหมาย% ในคำสั่ง modulo แล้วในภาษา SQL ยังใช้ฟังก์ชัน MOD แทนเครื่องหมาย % ได้ซึ่งจะให้ผลลัพธ์เช่นเดียวกัน

2.1.4.2 ตัวดำเนินการเปรียบเทียบ (Comparison Operators)

เป็น Operator ที่จะให้ค่าออกมา 3 ค่าคือ ถูก (TRUE) ผิด (FALSE)

2.1.4.3 ตัวดำเนินการอักขระ (Character Operators)

LIKE เป็นคำสั่งที่ใช้สำหรับการระบุเงื่อนไขการเลือกข้อมูลในตาราง โดยทำการค้นหาข้อความที่ระบุ ภายในฟิลด์ที่กำหนด ซึ่งอาจไม่ทราบค่าข้อมูลทั้งหมดที่จะค้นหาหรือรู้เพียงบางตัวอักษรเท่านั้นตัวดำเนินการ LIKE จะระบุต่อท้ายชื่อคอลัมน์ที่เป็นเงื่อนไขโดยจะใช้สัญลักษณ์ที่เป็นตัวค้นหาช่วยในการค้นหาข้อมูล สัญลักษณ์ดังกล่าวประกอบด้วย % ใช้แทนจำนวนอักษรได้หลายตัวและ _ (เครื่องหมายขีดเส้นใต้) ใช้แทนจำนวนที่ไม่ทราบค่า 1 ตัวโดยข้อมูลบางส่วนของใช้ในการค้นหาพร้อมกับสัญลักษณ์ทั้งสองนี้จะต้องมีเครื่องหมาย “ กำกับเสมอ

2.1.4.4 ตัวดำเนินการตรรกะ (Logical Operators)

เป็นตัวดำเนินการที่ให้ผลลัพธ์แค่ 2 ค่าเท่านั้น คือ True และ False มักจะใช้ในการตรวจสอบเงื่อนไข หรือสร้างเงื่อนไขขึ้นมาเพื่อทดสอบกรณีต่าง ๆ

2.1.4.4.1 ตัวดำเนินการ AND

เป็นตัวดำเนินการที่ใช้เชื่อมโยงค่า 2 ค่าโดยถ้าค่าหนึ่งเป็น TRUE อีกค่าหนึ่งเป็น TRUE จะให้ค่า TRUE ออกมาแต่ถ้าค่าหนึ่งเป็น TRUE อีกค่าหนึ่งเป็น FALSE จะให้ค่าเป็น FALSE

2.1.4.4.2 ตัวดำเนินการ OR

ใช้ในการเปรียบเทียบถ้าสิ่งทีนำมาเปรียบเทียบกับสิ่งใดสิ่งหนึ่งเป็นจริงจะได้ผลลัพธ์ออกมาเป็นจริง

2.1.4.4.3 ตัวดำเนินการ NOT

ในการเปรียบเทียบถ้าสิ่งทีนำมาเปรียบเทียบเป็นจริงจะได้ผลลัพธ์ออกมาเป็นเท็จแต่ถ้าสิ่งทีนำมาเปรียบเทียบเป็นเท็จผลที่ได้ออกมาจะเป็นจริงตัวดำเนินการ NOT ยังใช้กับ NULL ได้ถ้านำมา not กับ null รวมกันแล้วจะใช้สำหรับค่าที่ไม่ว่าง

2.1.4.4.4 ตัวดำเนินการ IN

เป็นการกำหนดเซตของสิ่งที่ต้องการค้นหาโดยการกำหนดชื่อของสมาชิกเซตลงไปในวงเล็บและแยกจากกันด้วยจุลภาค

2.1.4.4.5 ตัวดำเนินการ EXISTS

เป็นคำสั่งที่ใช้สำหรับการระบุเงื่อนไขโดยทำการตรวจสอบข้อมูลจากอีกตารางหนึ่งว่ามีข้อมูล หรือว่าไม่มีข้อมูลที่ต้องการเปรียบเทียบ

2.1.5 ฟังก์ชันที่ใช้ในภาษา SQL

เป็นฟังก์ชันซึ่งเก็บประจําไว้กับภาษา SQL มีฟังก์ชันอยู่ 6 ประเภทคือ

2.1.5.1 ฟังก์ชันในการรวม (Aggregate Functions)

เป็นกลุ่มฟังก์ชันที่ให้ผลของคำสั่งออกมาเพียง 1 คอลัมน์ซึ่งเป็นกลุ่มฟังก์ชันที่ใช้กับข้อมูลที่เป็นตัวเลขได้แก่

- 1) COUNT(x) ใช้นับจำนวนแถวของคอลัมน์(x)
- 2) SUM(x) ใช้หาผลรวมของคอลัมน์(x)
- 3) AVG(x) ใช้ในการหาค่าเฉลี่ยของคอลัมน์(x)
- 4) MAX(x) ใช้หาค่าสูงสุดของคอลัมน์นั้น(x)
- 5) MIN(x) ใช้ในการหาค่าต่ำสุดของคอลัมน์(x)
- 6) VARIANCE(x) ใช้หาค่าส่วนเบี่ยงเบนมาตรฐานยกกำลัง 2 ในคอลัมน์(x)
- 7) STDDEV(x) หาค่าส่วนเบี่ยงเบนมาตรฐานของคอลัมน์(x)

การใช้ฟังก์ชันในการรวมค่าต่างๆในภาษาSQL ดำเนินตามคำสั่งที่มีฟังก์ชันในการรวมค่าผลของคำสั่งจะแสดงค่าเพียงค่าเดียว

2.1.5.2 ฟังก์ชันตัวหนังสือ (Character functions)

เป็นฟังก์ชันที่ใช้สำหรับจัดการข้อมูลที่เป็นตัวอักษรโดยที่มีตัวแปรจริงเป็นชนิดอักขระหรือชนิดตัวเลขและให้ผลการคำนวณเป็นค่าอักขระหรือค่าตัวเลข

- 1) CHR(x) เป็นการเปลี่ยนนิพจน์อักขระให้เป็นรหัส ASCII ค่าที่ได้จะเป็นค่ารหัส ASCII
- 2) CONCAT(x, y) เป็นการรวมอักขระ(x และ y) เข้าด้วยกัน
- 3) INITCAP (<string>) เป็นการเปลี่ยนค่าตัวอักษร(string) ให้ตัวแรกเป็นอักขระตัวใหญ่แล้วตามด้วยอักขระตัวเล็ก
- 4) LOWER (<string>) and UPPER (<string>)

- 5) LOWER (<string>) เปลี่ยนอักษรเป็นตัวเล็ก
- 6) UPPER (<string>) เปลี่ยนอักษระตัวใหญ่
- 7) REPLACE (<string>, x, y) เป็นการแทนค่าอักษรx โดยการค้นหาตัวอักษรที่ต้องการแทนที่ แล้วแทนที่ด้วยอักษรy ที่ต้องการ
- 8) SUBSTR (<string>, x, y) เป็นการตัดตัวอักษร (<string>) ในตำแหน่งที่ x จำนวน y ตัว

2.1.5.3 ฟังก์ชันเชิงคำนวณ (Arithmetic functions)

- 1) ABS(x) เป็นการหาค่าสมบูรณ์ของคอลัมน์(x)
- 2) CEIL(x) AND FLOOR(x) เป็นการปัดค่าเลขจำนวนเต็มหรือเลขทศนิยม
- 3) COS(x), COSH(x), SIN(x), TAN(x) เป็นฟังก์ชันทางตรีโกณมิติ
- 4) EXP(x) เป็นการหาค่า e ยกกำลัง x
- 5) MOD(x, y) เป็นการเศษที่เกิดจาก x หารด้วย y
- 6) POWER(x, y) หาค่ายกกำลังโดย x เป็นเลขฐาน y เป็นเลขยกกำลัง
- 7) SQRT(x) เป็นการหารากที่ 2 ของข้อมูลในคอลัมน์ x

2.1.5.4 ฟังก์ชันการแปลง(Conversion functions)

TO_CHAR(X) จะทำการแปลง Data Type ให้เป็นตัวอักษร

2.1.5.5 ฟังก์ชันวันและเวลา(Date and time functions)

เป็นกลุ่มฟังก์ชันที่แสดงข้อมูลออกมาเป็นวันและเวลา

- 1) ADD_MONTHS(x, y) เป็นฟังก์ชันที่ต้องการบวกจำนวนเดือน (y) เข้าไปในข้อมูลคอลัมน์ (x)
- 2) LAST_DAY(x) เป็นฟังก์ชันที่แสดงวันสุดท้ายของเดือนในคอลัมน์(x)
- 3) MONTHS_BETWEEN(x, y) เป็นฟังก์ชันที่คำนวณค่าระหว่างx และy โดยมีหน่วยเป็นเดือน

2.1.5.6 การเรียกดูข้อมูลในรูปแบบต่างๆ

2.1.5.6.1 การเรียกดูข้อมูลโดยใช้ฟังก์ชันในการรวม

การเรียกดูข้อมูลโดยใช้ฟังก์ชันมีรูปแบบไวยากรณ์ดังต่อไปนี้

SELECT<column 1, column 2,...>

FROM <table name>

[WHERE<condition>]

[GROUP BY< grouping column>...]

[HAVING<condition>];

SELECT

Column 1, column 2,...

FROM

Table name

WHERE<condition>

GROUP BY < group column>

HAVING<condition>

คำสั่งที่ต้องมีทุกครั้งที่ต้องการเรียกค้นข้อมูล

คอลัมน์ที่ต้องการเรียกค้น

การกำหนดว่าให้เรียกดูข้อมูลได้จากตารางใด

ชื่อตารางที่ต้องการเรียกค้นข้อมูล

ส่วนของคำสั่งที่บอกเงื่อนไขที่จะใช้ในการค้นหาข้อมูล

ส่วนของคำสั่งที่บอกเงื่อนไขการจัดกลุ่ม

ใช้ควบคู่กันกับ GROUP BY เสมอเพื่อต้องการให้ได้

ข้อมูลที่จัดกลุ่มตาม GROUP BY

2.2 ภาษา PL/SQL (Procedural programming language)

PL/SQL เป็น Procedural programming language เป็นเครื่องมือใช้พัฒนาระบบงานที่เพิ่มความสามารถให้กับ SQL ที่ถูกพัฒนาขึ้นโดย Oracle ทำให้การพัฒนาระบบงานที่ซับซ้อนและการเข้าจัดการข้อมูลในฐานข้อมูลได้เป็นอย่างดีส่วนการทำงานจะยังสามารถใช้คำสั่ง SQL ได้เหมือนเดิม แต่จะมีลักษณะการทำงานเป็นแบบ Procedure หรือการรวมคำสั่ง SQL แต่ละ Statement ไว้เป็นชุดคำสั่งหนึ่งแล้วเรียกใช้งานทำให้คำสั่ง SQL มีประสิทธิภาพ และทำงานได้ตรงตามความต้องการได้มากขึ้นในกรณีที่ระบบมีความซับซ้อน PL/SQL ไม่ได้รับการออกแบบให้เป็นภาษาเดี่ยว (standalone language) แต่ได้รับการออกแบบมาเพื่อให้เรียกใช้จากในเครื่องแม่ข่ายของตัวเอง (host environment) การเขียนโปรแกรมเพื่อใช้งานฐานข้อมูล

Oracle ในส่วนของ PL/SQL จะมีประสิทธิภาพสูงสิ่งที่สำคัญคือ PL/SQL ได้รับการออกแบบมาให้เป็นอันหนึ่งอันเดียวกันกับ SQL โดยไม่ต้องขึ้นอยู่กับโปรแกรมตัวกลาง(Intermediate Software) ใดๆเพื่อที่จะทำการดำเนินการ SQL ในโปรแกรมแต่สามารถใช้คำสั่ง INSERT หรือ UPDATE ในโปรแกรมได้โดยตรง คำสั่งของ PL/SQL ถูกจัดรวมเป็นกลุ่มเรียกว่าบล็อกบล็อกเป็นหน่วยโปรแกรมที่สามารถทำงานได้และทั้งเป็นขอบเขตของตัวแปรที่ประกาศด้วยถ้าใช้ PL/SQL เขียน Procedure หรือ Package ก็จะมีการตั้งชื่อให้กับบล็อกนั้นๆถ้าไม่มีการตั้งชื่อก็จะเรียกว่า Anonymous Block

2.2.1 โครงสร้างของ PL/SQL Block

หน่วยคำสั่งของ PL/SQL ประกอบด้วยบล็อกซึ่งอาจมีหนึ่งบล็อกหรือมากกว่าก็ได้บล็อกของ PL/SQL ประกอบด้วยสามส่วนดังต่อไปนี้

DECLARE – Optional

– Variables, cursors, user-defined exception

BEGIN – Mandatory

– SQL statements or

– PL/SQL statements

EXCEPTION – Optional

– Action to perform when errors occur

END; – Mandatory

หน่วยคำสั่งพื้นฐานของ PL/SQL คือกระบวนการคำสั่ง (Procedures) และ Functions (ทั้ง 2 หน่วยคำสั่งเป็น Subprogram) และ Anonymous Block

- 1) Anonymous Block คือบล็อกที่ไม่มีชื่อและสั่งให้ทำงานได้ทันทีหรือเป็นส่วนที่ฝังอยู่ใน pre-compiler program
- 2) Subprograms คือบล็อกของ PL/SQL ที่มีชื่อกำกับซึ่งอาจจะประกอบด้วยตัวแปรเสริม (Parameter) และสามารถเรียกใช้งานได้เช่นเดียวกับภาษาระดับสูงอื่นๆซึ่งสามารถสร้าง Subprogram ในรูปของ Procedure หรือ Function

PL/SQL Block Types

Anonymous	Procedure	Function
<pre> DECLARE BEGIN -statements EXCEPTION END;</pre>	<pre> PROCEDURE <name> IS BEGIN -statements EXCEPTION END;</pre>	<pre> FUNCTION <name> RETURN <datatype> IS BEGIN -statements EXCEPTION END;</pre>

รูปที่ 2.1 ชนิดของ Block ใน PL/SQL

2.2.2 Identifiers คือชื่อที่ตั้งให้แก่ PL/SQL Object ต่างๆดังต่อไปนี้

- 1) Constant or variable
- 2) Exception
- 3) Cursor
- 4) Program name: procedure, function, package, object type, trigger, etc.
- 5) Reserved word
- 6) Label

2.2.2.1 คุณสมบัติของ PL/SQL Identifiers

- 1) มีขนาดความยาวไม่เกิน 30 ตัวอักษร
- 2) ต้องเริ่มต้นด้วยตัวอักษร
- 3) สามารถประกอบด้วยอักษร \$ (dollar sign), _ (underscore), และ # (pound sign) ได้

4) จะต้องไม่มีการเว้นช่องว่าง ("whitespace" characters) เป็นส่วนประกอบ

2.2.2.2 PL/SQL ประกอบด้วย Built-In Identifiers 2 ชนิด

- 1) Language Keywords
- 2) Identifiers จากใน Standard Package

2.2.2.3 Semicolon Delimiter

คำสั่งและคำประกาศใน PL/SQL ปิดท้ายด้วยอัฒภาค (;) ความจริงแล้วคำสั่งหนึ่งอาจจะกระจายอยู่หลายบรรทัดได้เพื่อให้อ่านได้สะดวก

2.2.2.4 Comment ใน PL/SQL มีใช้สองแบบคือ

- 1) แบบบรรทัดเดียวจะใช้เครื่องหมายขีดสองตัวติด (--) กัน บรรทัดนั้นจะกลายเป็น Comment
- 2) Comments แบบหลายบรรทัดเริ่มต้นด้วย (/*) และปิดท้ายด้วย (*/) ซึ่งความหมายและการใช้งานก็เหมือนกันกับที่ใช้ใน C++ หรือ JAVA

2.2.3 การประกาศตัวแปรและการใช้ตัวแปร

การประกาศตัวแปรจะต้องทำในส่วนการประกาศตัวแปร (Declaration) ของบล็อกและสามารถกำหนดค่าเริ่มต้นได้และการใช้งานตัวแปรก็มีลักษณะเดียวกับตัวแปรในภาษาโปรแกรมระดับสูงอื่นๆชนิดของตัวแปรที่ใช้ได้ใน PL/SQL คือ

- 1) Scalar เป็นชนิดตัวแปรที่เก็บค่า ณ เวลาหนึ่งปกติจะเป็นชนิดเดียวกันกับสคาลาร์ในตารางของฐานข้อมูลนอกจากนั้น PL/SQL ยังสนับสนุนตัวแปรชนิด Boolean เช่นกันประเภทข้อมูลแบบมีลำดับ (Scalar Data Type) ประกอบด้วยตัวแปรชนิด VARCHAR2, NUMBER, DATETIME, CHAR, LONG, BOOLEAN, BINARY_INTEGER หรือ PLS_INTEGER, BINARY_FLOAT, BINARY_DOUBLE เป็นต้น
- 2) Composite เช่นระเบียน (record) ของตาราง
- 3) Reference หรือ Pointer
- 4) LOB หรือ Locators เป็นตัวกำหนดตำแหน่งของ Object ขนาดใหญ่เช่นรูปภาพ

5) Non-PL/SQL กลุ่มตัวแปรในภาษาหลัก (Host Language) ในโปรแกรมที่ใช้ก่อนแปลโปรแกรม (Pre-Compiler Program) หรือ SQL*Plus Variables หรือตัวแปรในช่องข้อมูลของ FORM

2.2.3.1 Records Type

ในกรณีที่ต้องการใช้ตัวแปรเพื่อรับค่าจากคอลัมน์ของตารางจะต้องให้ตัวแปรนั้นเป็นชนิดเดียวกับคอลัมน์ด้วยหรือต้องการให้ตัวแปรที่ประกาศนั้นเป็นตัวแปรชนิดเดียวกับตัวแปรอีกตัวหนึ่งก็สามารถทำได้ด้วยการใช้ %TYPE โดยให้นำหน้า %TYPE ด้วยชื่อตารางและคอลัมน์ เช่นเดียวกับ %ROWTYPE สามารถเก็บตัวแปรชนิดตามตารางได้ เมื่อใช้ ROWTYPE จะเรียก คอลัมน์ในตารางมาอ้างอิงในการเก็บ

```
v_name      emp.ename%TYPE;
v_sal       NUMBER (7, 2);
v_comm      emp%ROWTYPE := 10;
v_comm.empno := 50;
```

ตัว NOT NULL ใช้ไม่ได้กับตัวแปรที่ประกาศโดยใช้ %TYPE

2.2.3.2 Composite Data Types

คือตัวแปรชนิด Composite คือ TABLE, RECORD, NESTED TABLE และ VARRAY

2.2.3.3 LOB Data Types

คือ Large Object (LOB) ใช้เก็บบล็อกของข้อมูลแบบไร้โครงสร้าง (Unstructured Data) เช่น Text, Graphic Images, Video Clip, Sound Wave Forms ที่มีขนาดได้ถึง 4 Gigabytes ซึ่งประกอบด้วย

- 1) Character large object (CLOB) บล็อกของข้อมูลแบบ single-byte character
- 2) Binary large object (BLOB) ข้อมูลแบบ binary
- 3) Binary file (BFILE) ใช้ในการจัดเก็บ binary object ของ operating system files นอกฐานข้อมูล
- 4) National language character large object (NCLOB) เป็น CLOB ประเภทหนึ่ง แต่ support พวง UniCode character ด้วย

2.2.3.4 Bind Variable

คือตัวแปรที่ประกาศอยู่ในสภาพแวดล้อมของ Host และใช้เพื่อส่งผ่านค่าในขณะใช้งาน (Runtime) เข้าในหรือออกจากโปรแกรม PL/SQL ซึ่งใช้งานได้เหมือนตัวแปรปกติเช่น

```
SQL>VARIABLE return_code NUMBER
```

```
SQL>VARIABLE return_msgVARCHAR2(30)
```

2.2.3.5 การกำหนดค่าให้ตัวแปร

ในการอ้างถึงตัวแปรหลักในส่วนของโปรแกรมที่ใช้อยู่ (Host Variable) ต้องใช้เครื่องหมายหัพภาคคู่ (:) นำหน้าตัวแปรเพื่อเป็นตัวแยกตัวแปรของ host ออกจากตัวแปรที่ประกาศใน PL/SQL เช่นถ้าตัวแปร p_sal เป็นตัวแปรที่ประกาศด้วย VARIABLE แล้วการกำหนดค่าให้ตัวแปรนี้ในบล็อก PL/SQL เขียนได้เป็น :p_sal := v_sal / 12;

2.2.4 การใช้คำสั่ง SQL ภายในบล็อกของ PL/SQL

เมื่อต้องการดึงข้อมูลหรือเปลี่ยนแปลงข้อมูลในฐานข้อมูลจะต้องใช้คำสั่งของ SQL และ PL/SQL ซึ่งต้องคำนึงถึงดังนี้

2.2.4.1 คำสั่ง SELECT ใน PL/SQL

การใช้คำสั่ง SELECT ภายในบล็อกของ PL/SQL ในการดึง Record จากตาราง ฐานข้อมูลมีรูปแบบคำสั่งดังนี้

```
SELECT expression INTO { variable [, variable]... | record_name}
```

```
FROM tables WHERE condition;
```

Expression	เป็นรายการอย่างน้อย 1 คอลัมน์และอาจประกอบด้วยนิพจน์ของ SQL หรือฟังก์ชันที่ใช้กับข้อมูลเป็นแถว (Row Function) หรือฟังก์ชันที่ใช้กับข้อมูลเป็นกลุ่ม (Group Function)
Variable	เป็นตัวแปรแบบ Scalar เพื่อเก็บค่าที่ดึงได้
Record_name	เป็น PL/SQL Record เพื่อเก็บค่าที่ดึงได้
Tables	ชื่อตารางที่ต้องการดึงข้อมูลมา
Condition	ประกอบด้วยชื่อคอลัมน์นิพจน์ค่าคงที่และการเทียบค่ารวมทั้งตัวแปรและค่าคงที่

ของ PL/SQL ได้

ข้อจำกัดของการใช้ SELECT ... INTO เพื่อดึงข้อมูลจาก database คือ การ SELECT จะต้องได้ข้อมูล 1 record เท่านั้น (ไม่เจอก็ไม่ได้ จะเกิด NO_DATA_FOUND exception และถ้าเจอมากกว่า 1 record ก็เกิด TOO_MANY_ROWS exception) หากต้องการจะ SELECT ข้อมูลที่หลาย records จะต้องใช้ Explicit Cursor ช่วย

2.2.4.2 คำสั่ง DML ใน PL/SQL

สามารถใช้คำสั่ง DML (INSERT, UPDATE, และ DELETE) ในบล็อก PL/SQL ได้โดยไม่มีข้อจำกัดใดๆ รวมทั้ง COMMIT และ ROLLBACK เช่น

```
BEGIN
INSERT INTO EMP (empno, ename, job, deptno)
VALUES (empno_sequence.NEXTVAL, 'HARDING', 'CLERK', 10);
END;
```

2.2.5 คำสั่งโครงสร้างการเลือก (Selection Statement)

คำสั่งการเลือกของ PL/SQL ประกอบด้วยรูปแบบของคำสั่ง IF ดังนี้

- 1) IF-THEN-END IF
- 2) IF-THEN-ELSE-END IF
- 3) IF-THEN-ELSIF- END IF

คำสั่ง IF มีรูปแบบดังนี้

```
IF condition THEN
    Statement;
[ELSIF condition THEN
    Statement;]
[ELSE
```

```
Statement;]
END IF;
```

เมื่อ Condition เป็น Boolean expression ที่ให้ผลเป็น TRUE, FALSE หรือ NULL

คำสั่ง IF อาจจะใช้อยู่ในรูปแบบซ้อนได้ (nested) ดังนี้

```
IF condition1 THEN
    statement1;
ELSE
    IF condition2 THEN
        statement2;
    END IF;
END IF;
```

หรือรูปแบบ IF-THEN-ELSIF ดังนี้

```
IF condition1 THEN
    statement1;
ELSIF condition2 THEN
    statement2;
ELSIF condition3 THEN
    statement3;
END IF;
```

2.2.6 สร้างโครงสร้างการวนรอบ (Iteration)

PL/SQL ประกอบด้วยคำสั่งการทำงานวนรอบเพื่อทำงานซ้ำเช่นเดียวกับภาษาโปรแกรมระดับสูงอื่นๆเช่นกันคือ

- 1) คำสั่ง LOOP
- 2) คำสั่ง FOR
- 3) คำสั่ง WHILE

และใช้คำสั่ง EXIT เพื่อบังคับให้ออกจากการวนรอบทำงานนอกจากนี้ยังมีคำสั่ง FOR ...LOOP สำหรับ Cursor

คำสั่ง LOOP มีรูปแบบดังนี้

```

statement1;           -- statements
...
EXIT [WHEN condition]; -- exit statement
END LOOP;             -- delimiter

```

เมื่อ Condition เป็นนิพจน์ที่ให้ค่าเป็น Boolean กรณีที่เกิดการทำงานของคำสั่ง EXIT ขึ้นจะทำให้การทำงานออกไปทำคำสั่งที่อยู่ต่อจาก END LOOP; การทำงานในการวนซ้ำจะเกิดขึ้นหนึ่งครั้งเสมอเป็นอย่างน้อยถ้าคำสั่ง EXIT ภายใน Loop อยู่ทางด้านท้ายของ Loop แต่ถ้าคำสั่ง EXIT อยู่ด้านบนสุดของ Loop ต่อจาก END LOOP ก็อาจจะไม่มีการทำงานคำสั่งอื่นๆที่ตามมา

คำสั่ง FOR มีรูปแบบดังนี้

```

FOR counter IN [REVERSE] lower_bound..upper bound LOOP
statement1;
statement2;
...
END LOOP;

```

เมื่อตัวแปร counter เป็นตัวนับที่เป็นเลขจำนวนเต็มที่เพิ่มขึ้นหรือลดลง (เมื่อใช้ REVERSE) ครั้งละ 1 โดยอัตโนมัติและจะมีการทำงานตามคำสั่งใน Loop หนึ่งรอบเมื่อเปลี่ยนค่าไปหนึ่งครั้งซึ่งจะทำการจำนวนครั้งจนกระทั่งมันมีค่าถึงขอบเขตบน (Upper Bound) หรือขอบเขตล่าง (Lower Bound) และไม่จำเป็นต้องประกาศตัวแปร counter เนื่องจากจะประกาศให้อย่างอัตโนมัติตัวอย่างคำสั่งข้างล่างนี้มีการทำงาน statement1 เพียงหนึ่งครั้ง

คำสั่ง WHILE มีรูปแบบคำสั่งดังนี้

WHILE condition LOOP

statement1;

statement2;

...

END LOOP;

จะทำงานวนรอบซ้ำภายใน Loop ตราบเท่าที่ Condition ยังคงเป็น TRUE และจะหยุดทำงานเมื่อ Condition เป็น FALSE หรือ NULL และ statement เป็นคำสั่ง PL/SQL หรือ SQL

Nested Loop และ Labels สามารถเขียนคำสั่งทำงานวนรอบแบบซ้อนกันได้ (Nested loop) โดยที่คำสั่งที่ซ้อนกันนั้นจะใช้ LOOP FOR หรือ WHILE ซ้อนกันได้กำหนด Label ให้กับคำสั่งทำงานวนรอบและสามารถสั่งให้ออกจากคำสั่งทำงานวนรอบด้านนอกได้ด้วยการใช้คำสั่ง EXIT

ชื่อของ Label กำหนดอยู่หน้าคำสั่งที่อาจจะอยู่บรรทัดเดียวหรือคนละบรรทัดกับคำสั่งก็ได้โดยเขียนอยู่ในวงเล็บเหลี่ยมสองตัว<<label>>

...

BEGIN

<<Outer_loop>>

LOOP

v_counter := v_counter - 1;

EXIT WHEN v_counter > 10;

<<Inner_loop>>

LOOP

...

EXIT Outer_loop WHEN total_done = 'YES';

-- Leave both loop

EXIT WHEN Inner_done = 'YES';

-- Leave inner loop only

...

END LOOP Inner_loop;

```

...
END LOOP Outer_loop;
END;

```

2.3 JavaScript

JavaScript เป็นภาษาสคริปต์เชิงวัตถุหรือเรียกว่าอ็อบเจกต์โอเรียนเต็ลเต็ด (Object Oriented Programming) ที่มีเป้าหมายในการออกแบบและพัฒนาโปรแกรมในระบบอินเทอร์เน็ตสำหรับผู้เขียนเอกสารด้วยภาษา HTML สามารถทำงานข้ามแพลตฟอร์มได้ทำงานร่วมกันกับภาษา HTML และภาษาจาวาได้ทั้งทางฝั่งลูกข่าย (Client) และทางฝั่งแม่ข่าย (Server) โดยมีลักษณะการทำงานดังนี้

- 1) Navigator JavaScript เป็น Client-Side JavaScript ซึ่งหมายถึง JavaScript ที่ถูกแปลทางฝั่งไคลเอนต์จึงมีความเหมาะสมต่อการใช้งานของผู้ใช้ทั่วไปเป็นส่วนใหญ่
- 2) Live Wire JavaScript เป็น Server-Side JavaScript ซึ่งหมายถึง JavaScript ที่ถูกแปลทางฝั่งเซิร์ฟเวอร์

2.3.1 คำสั่งการเขียนสคริปต์ของ Java Script เริ่มต้นด้วย

```

<script language="JavaScript">
-----
-----
</script>

```

2.3.2 การประกาศตัวแปร(Declaring variables)

ใน Javascript การประกาศตัวแปร จะทำทั้งหมด 3 ขั้นตอนคือ

- 1) Declaration : คือการประกาศชื่อตัวแปร เช่น var name;
- 2) Assignment : คือขั้นตอนการกำหนดค่า เช่น name = 'Chai';

ในกรณีที่ต้องการกำหนดตัวแปรหลายตัวในบรรทัดเดียวกันให้ใช้เครื่องหมายจุลภาค (,) คั่นระหว่างชื่อตัวแปรและปิดท้ายด้วยเครื่องหมายอัฒภาค (;)

2.3.3 การสร้าง function

กลุ่มของโค้ดที่คุณสามารถเรียกใช้ได้ทุกที่ในโปรแกรมจะขึ้นต้นด้วยคำว่า function

การเขียนไวยากรณ์สำหรับสร้างฟังก์ชันนั้นมีดังนี้

```
function functionname(parameter-list)
{
    statements
}
```

- 1)function name ชื่อของฟังก์ชัน
- 2)parameter-list ชื่อของพารามิเตอร์อ้างอิง (Argument) แล้วส่งผ่านไปยังฟังก์ชัน
ต้องระบุอยู่ในเครื่องหมายวงเล็บเท่านั้น โดยจะมีพารามิเตอร์เพียงตัวเดียวหลายตัวหรือไม่มีเลย
กรณีที่มีพารามิเตอร์หลายๆตัวแต่ละตัวจะต้องเขียนแยกออกจากกันด้วยเครื่องหมายจุลภาค (,)
- 3)statements กลุ่มของโค้ดที่ทำงานภายในฟังก์ชัน

2.3.4 ชนิดข้อมูลของตัวแปร

- 1) Number หมายถึงข้อมูลชนิดตัวเลขประกอบด้วยเลขจำนวนเต็ม (Integer) และเลขจำนวนจริง (Float)
- 2) Logical หมายถึงข้อมูลทางตรรกะมี 2 สถานะคือจริง (True) และเท็จ (False)
- 3) String หมายถึงข้อมูลที่เป็นข้อความซึ่งจะต้องกำหนดไว้ในเครื่องหมายคำพูด ("...")
- 4) Null หมายถึงไม่มีค่าข้อมูลใดๆซึ่งค่า Null ใช้สำหรับการยกเลิกพื้นที่เก็บค่าของตัวแปรออกจากหน่วยความจำ

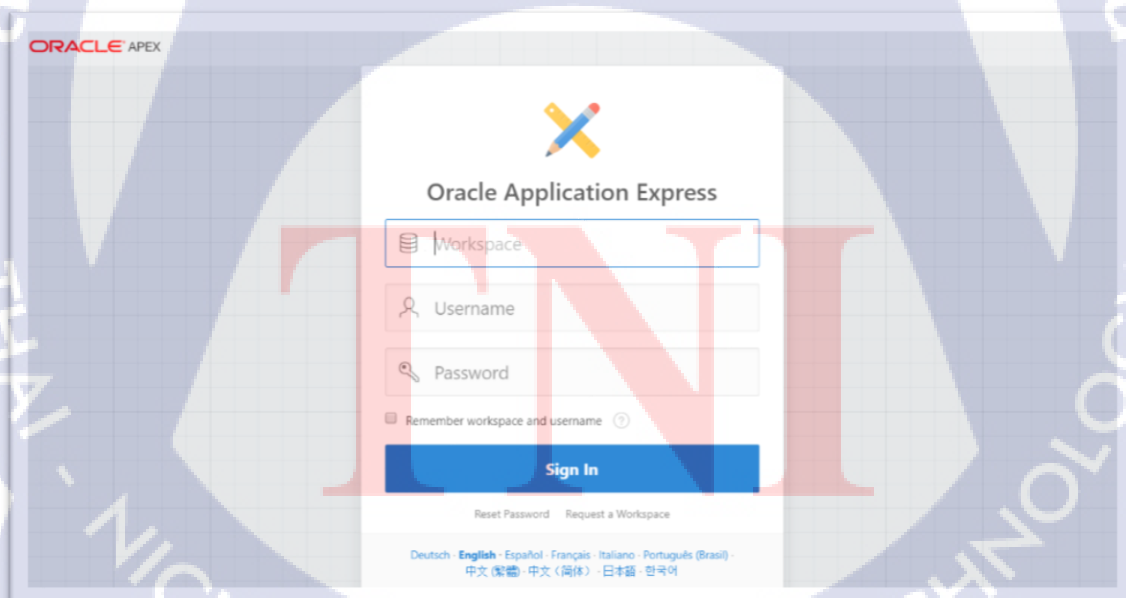
2.3.5 Reserved Words

- 1) ชื่อของตัวแปร ต้องไม่ซ้ำกับคำสงวน
- 2) ชื่อของตัวแปรต้องไม่ขึ้นต้นด้วยตัวเลข ควรขึ้นต้นด้วยตัวอักษร หรือขีดเส้นใต้
- 3) ชื่อของตัวแปร เป็นแบบ case sensitive นั่นคือ ตัวใหญ่กับตัวเล็กถือว่าเป็นตัวต่างกัน

abstract	else	instanceof	switch
boolean	enum	int	synchronized
break	export	interface	this
byte	extends	long	throw
case	false	native	throws
catch	final	new	transient
char	finally	null	true
class	float	package	try
const	for	private	typeof
continue	function	protected	var
debugger	goto	public	void
default	if	return	volatile
delete	implements	short	while
do	import	static	with
double	in	super	

รูปที่ 2.2 JavaScript Reserved Words

2.4 Oracle Application Express (Oracle APEX)

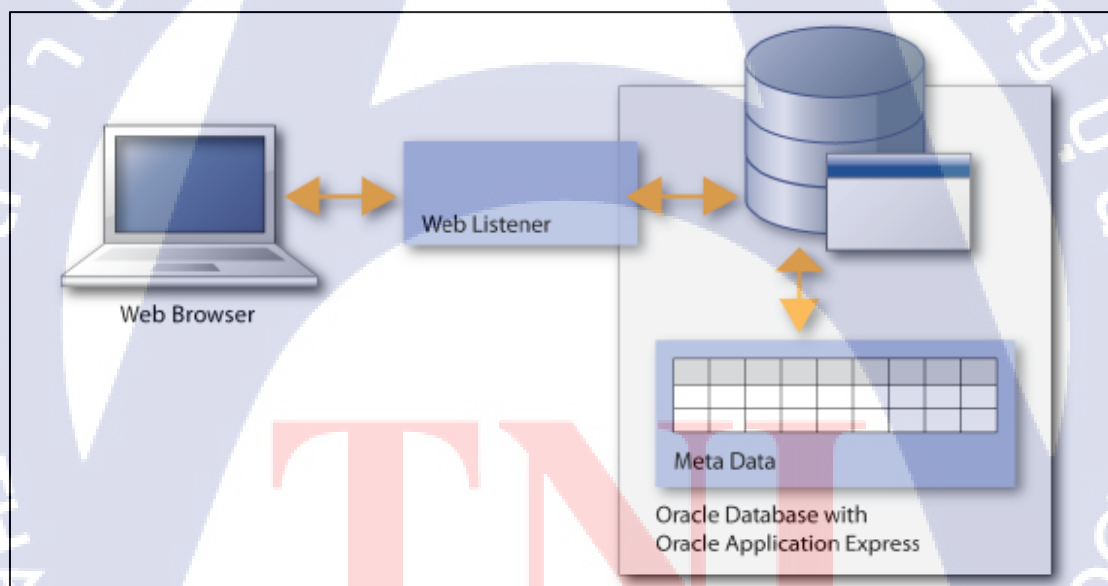


รูปที่ 2.3 Oracle Application Express (Oracle APEX)

Oracle APEX เป็นเครื่องมือพัฒนาระบบประยุกต์บนเว็บเบราว์เซอร์ (Web Application) ที่รวดเร็วสำหรับฐานข้อมูล Oracle เพียงใช้ผ่านเว็บเบราว์เซอร์กับประสบการณ์การเขียนโปรแกรมที่จำกัดแต่สามารถพัฒนาและปรับการใช้งานระบบประยุกต์ (Application) อย่างรวดเร็วและปลอดภัยได้

Oracle APEX มีการพัฒนาอย่างต่อเนื่องเป็นตัวเลือกในฐานข้อมูล Oracle และไม่มีค่าใช้จ่ายซื้อเดิมของ Oracle APEX นั่นคือ HTML DB และรองรับ Oracle Database 10.2.0.3 ขึ้นไปรวมถึง Oracle Database 10g Express Edition (Oracle XE) การติดตั้ง Oracle APEX จะเป็นส่วนหนึ่งของการติดตั้ง Oracle Database 11g

Rapid Application Development (RAD) ด้วยตัวช่วยที่ง่ายและการเขียนโปรแกรมแบบ Declarative (ไม่มี code ถูกสร้างหรือ Compile) ช่วยให้สามารถสร้างรายงานและโปรแกรมรับค่าได้อย่างมีประสิทธิภาพ สามารถสร้างรายงานจาก Spread Sheet (Upload) หรือจากตารางข้อมูลและตารางข้อมูลจำลองในฐานข้อมูล Oracle



รูปที่ 2.4 APEX Architecture

นิยามได้ว่า Web Browser หรือ User ต้องการข้อมูลที่ต้องการโดยจะมีตัวรับฟังหรือตัวกลางคือ Web Listener ที่จะรับข้อมูลของ User ส่งเข้า Database เพื่อมา Query และส่งกลับไปให้ User

2.4.1 การพัฒนาโปรแกรมแบบDeclarative

บน APEX การใช้งานจะเป็นแบบ Declarative คือไม่ต้องสร้างหรือแปลโปรแกรมตัวอย่างของโปรแกรม Declarative ใน Oracle APEX เช่น

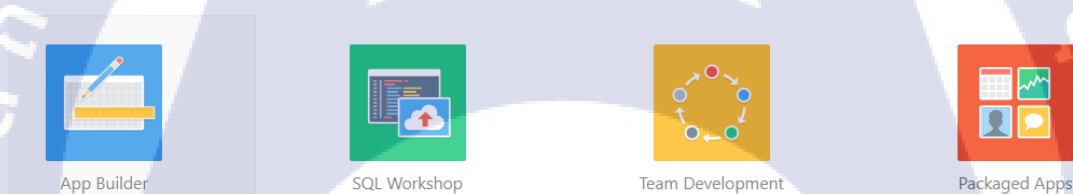
2.4.1.1 Interactive Reports

ผู้พัฒนาสามารถเลือกตารางข้อมูลหรือกำหนด SQL Statement ดังนั้นสามารถจำกัดความสามารถของผู้ใช้เช่นปิดการ Download หรือไม่อนุญาตในบางคอลัมน์ด้วยการซ่อนไว้

2.4.1.2 Interactive Grid

ผู้พัฒนาสามารถเลือกตารางข้อมูลหรือกำหนด SQL Statement เช่นเดียวกับ Interactive Reports แต่ Grid สามารถปรับแต่งขนาดคอลัมน์ได้

2.4.2 องค์ประกอบ APEX



รูปที่ 2.5 องค์ประกอบ APEX

2.4.2.1 App Builder

- 1) App Builder ใช้เพื่อให้ผู้พัฒนา Application ใช้ตัวช่วยเพื่อประกอบงานเป็น page ขึ้นมาโดยแต่ละ page นั้นจะประกอบด้วย Report, Chart, Map, Calendar, Form เป็นต้น
- 2) From a Spreadsheet คือตัวช่วยให้ผู้พัฒนาสามารถอัปโหลดข้อมูลนามสกุล .XLSเพื่อสร้างฐานข้อมูลอัตโนมัติ

2.4.2.2 SQL Workshop

ใช้ในการจัดการและดูแล Object ในฐานข้อมูล

2.4.2.3 Team Development

ช่วยจัดการวงจรของการพัฒนาโปรแกรมมีการติดตามและการจัดการของ Application, หา Bugs และความคิดเห็นผู้ใช้

2.4.2.4 Packaged Apps

ใช้ในการติดตั้งAppเสริม เพื่อสนับสนุนหรือแก้ไขปัญหา ปรับแต่ง App ให้ดียิ่งขึ้น

2.5 ข้อมูลระบบเงินเดือน

2.5.1 เงินได้

รายได้ต่างๆไม่ว่าจะเป็น เงินเดือน ค่าจ้าง ค่าไร ดอกเบี้ย หรือผลประโยชน์ต่าง ๆ ที่อาจคิดคำนวณเป็นภาษีเงินได้บุคคลธรรมดา

2.5.2 เงินหัก

การเสียภาษี โดยเงินรายได้จะถูกหักจากผู้จ่ายเพื่อนำส่งให้สรรพากร

2.5.3 ค่าลดหย่อน

สิทธิประโยชน์ทางภาษีอย่างหนึ่งที่จะช่วยให้เสียภาษีน้อยลงเมื่อคำนวณภาษี เช่น ค่าลดหย่อนน้ำ_ท่วม ค่าลดหย่อนประกันสังคม เป็นต้น

2.5.4 เงินล่วงเวลา

เงินที่นายจ้างจ่ายให้แก่ลูกจ้างเป็นการตอบแทนการทำงานล่วงเวลาในวันทำงาน

บทที่ 3
แผนงานการปฏิบัติงานและขั้นตอนการดำเนินงาน

3.1 แผนงานการฝึกงาน

ตารางที่ 3.1 แผนปฏิบัติงานโครงการ

หัวข้องาน	ม.ย.	ก.ค.	ส.ค.	ก.ย.
ศึกษาการใช้งานโปรแกรมที่จำเป็นต่อการพัฒนาระบบ				
ศึกษาหัวข้อน่าสนใจของระบบงานที่จะพัฒนาและจัดทำ POC				
พัฒนาระบบ Time Attendance				
ตรวจสอบความถูกต้อง และแก้ไขเมื่อพบปัญหา				
จัดทำเอกสารที่เกี่ยวข้อง				

3.2 รายละเอียดที่นักศึกษาปฏิบัติในการฝึกงาน

3.2.1 Oracle Application Express (APEX)

การปฏิบัติงานสหกิจศึกษา ข้าพเจ้าได้มีส่วนของ Payroll ของแผนก Business Solutions โดยได้เข้าร่วมในการพัฒนาระบบ Payroll เพื่อให้ระบบมีประสิทธิภาพยิ่งขึ้นนำเทคโนโลยีใหม่ หรือ หัวข้อที่น่าสนใจมาสนับสนุนและสามารถผ่านการทดสอบบนเครื่องDevนำมาใช้งานจริงบนเครื่องUATได้อีกทั้ง นำเสนอวิธีการใช้งานระบบPayroll สำหรับ User ที่ต้องการใช้งาน Apex

ซึ่งการกำหนดขอบเขตข้อมูลนั้น จะต้องเข้าใจว่าหน้าแต่ละหน้าใช้ข้อมูล Table ใด ทำการพัฒนาเครื่องใด จัดเก็บข้อมูลรูปแบบไหน และสามารถแสดงข้อมูลให้ถูกต้อง โดยสามารถอิง Data Dictionary ของบริษัทที่ออกแบบไว้ เป็นโครงร่างในการใช้ข้อมูลใน Table

3.2.2 งานอื่นๆ

3.2.2.1 จัดเตรียม Template ข้อมูลลูกค้า

3.2.2.2 ตรวจสอบข้อมูลลูกค้าให้อยู่ในรูปแบบที่ถูกต้อง

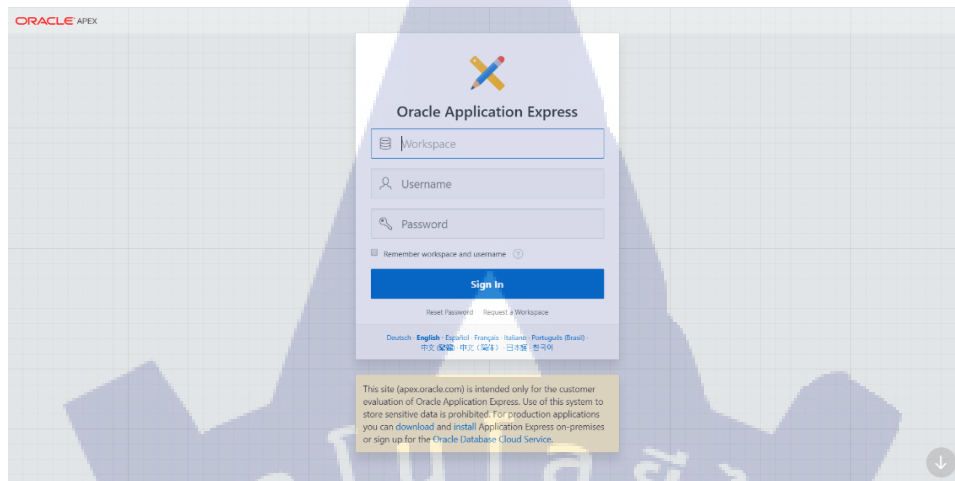
3.3 ขั้นตอนการดำเนินงานที่นักศึกษาปฏิบัติงาน

3.3.1 ศึกษาการใช้งานแอปพลิเคชันที่จำเป็นต่อการพัฒนาระบบPayroll

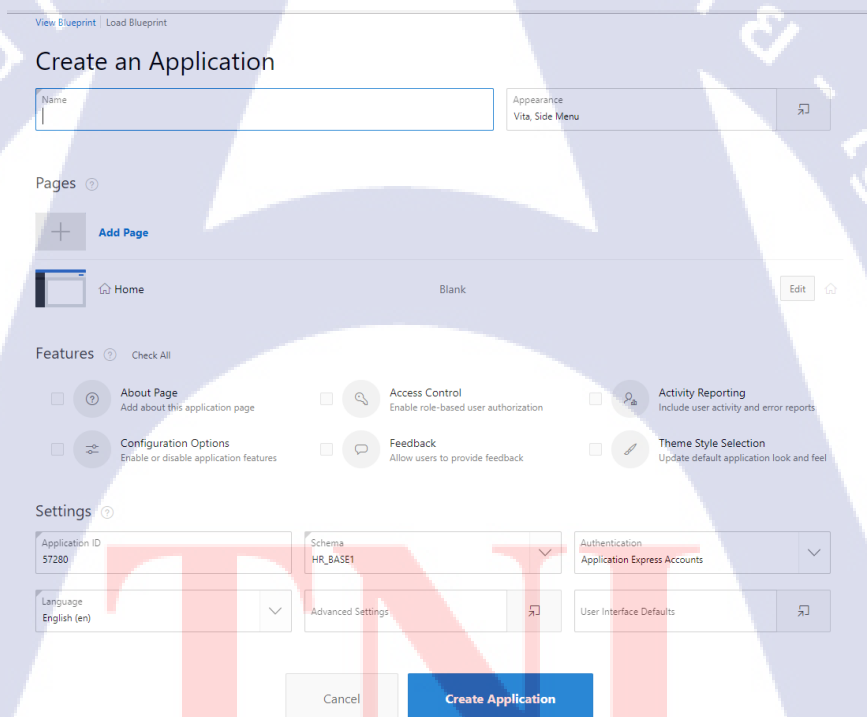
แอปพลิเคชันที่ทำการศึกษาการใช้งานคือ แอปพลิเคชัน Oracle Application Express (APEX) ซึ่งเป็นแอปพลิเคชันที่จำเป็นอย่างยิ่งต่อการพัฒนาระบบ Payroll เพราะเป็นแอปพลิเคชันที่ช่วยในการออกแบบ GUI และ เขียนโค้ดด้วยภาษา SQL / PL SQL ที่ใช้งานในระบบPayroll

3.3.2 การใช้งานแอปพลิเคชัน Oracle Application Express (APEX)

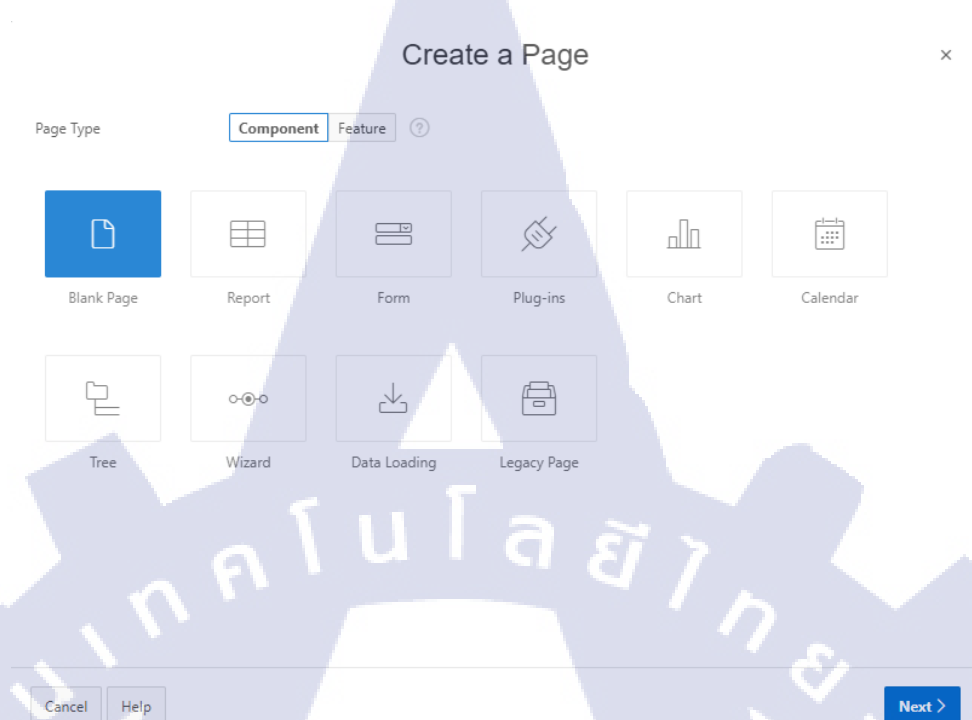
ขั้นตอนการใช้งานโดยจะมีกระบวนการตั้งแต่ การสร้าง / เลือก แอปพลิเคชันที่ต้องการ การสร้างหน้า Page ที่จำเป็นในการพัฒนาระบบ



รูปที่ 3.1 ตัวอย่างหน้าการใช้งานApex



รูปที่ 3.2 ตัวอย่างรายละเอียดการสร้างแอปพลิเคชัน



รูปที่ 3.3 ตัวอย่างรายละเอียดการสร้างหน้ารูปแบบต่างๆ

3.3.3 การสร้างฟังก์ชันและทำPOC

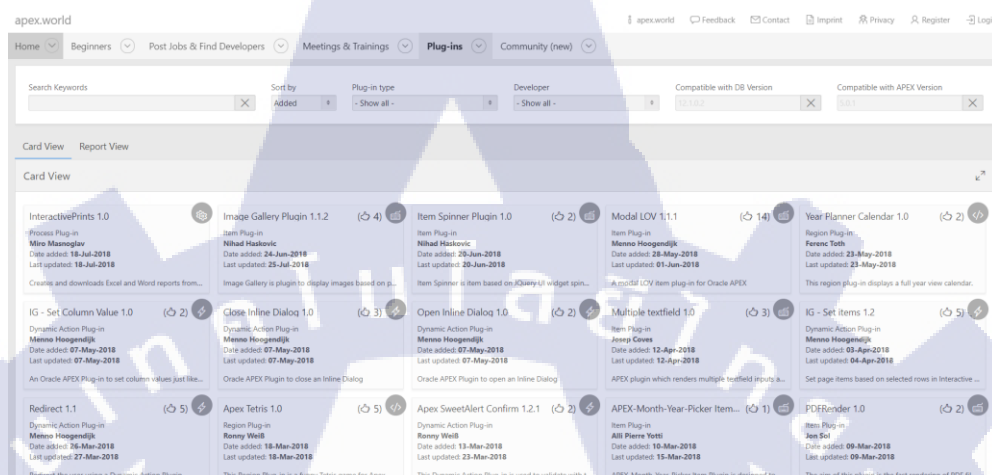
เมื่อทำการศึกษาหัวข้อที่น่าสนใจ นำมาสร้างฟังก์ชันด้วยภาษา SQL , PL/SQL เพื่อสนับสนุนการใช้งานในระบบ Payroll ให้มีประสิทธิภาพยิ่งขึ้น และทำ POC เพื่อทดสอบการใช้งาน ก่อนจะดำเนินงานจริง



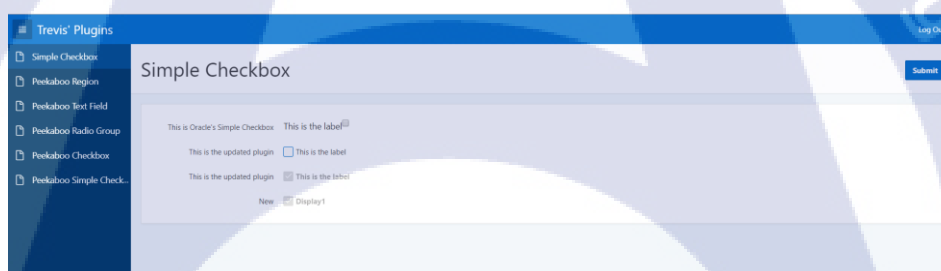
รูปที่ 3.4 ตัวอย่างการทดสอบเบื้องต้น(POC)

3.3.4 การเสริม Plug-in

การเสริมปลั๊กอินมาช่วยในเรื่องของการทำงาน เพิ่มความสามารถให้กับระบบ Payroll เพิ่มลูกเล่นต่างๆ ซึ่งสามารถดาวน์โหลดได้จาก <https://apex.world>



รูปที่ 3.5 ตัวอย่างหน้าดาวน์โหลด Plug-in



รูปที่ 3.6 ตัวอย่างการเทียบ Plug-in

3.3.5 ตรวจสอบความถูกต้องและแก้ไขปัญหา

หลังจากทดสอบเบื้องต้นหากพบปัญหานั้นมาเป็นหัวข้อเพื่อแก้ไขปัญหามาให้พร้อมสำหรับการใช้งานครั้งต่อไป หรือ สิ่งที่ต้องเพิ่มจากการได้ Feedback นำมาปรับปรุงระบบ Payroll ให้ดียิ่งขึ้น

3.3.6 จัดการทำเอกสารที่เกี่ยวข้อง

จัดทำเอกสารประกอบการเขียนฟังก์ชันเพื่อใช้เป็นเอกสารอ้างอิงให้แก่บริษัท

บทที่ 4

สรุปผลการดำเนินงาน การวิเคราะห์และสรุปผลต่าง ๆ

4.1 ขั้นตอนและผลการดำเนินงาน

ในช่วงระยะเวลาสี่เดือนของการปฏิบัติงานสหกิจศึกษา ได้มีการดำเนินการตั้งแต่เริ่มต้นจนปัจจุบันอยู่ในระหว่างการพิจารณา ปรับปรุง แก้ไขและเพิ่มเติม เพื่อเตรียมความพร้อมสำหรับการใช้งานบนระบบจริง

โดยได้รับคำแนะนำจากพนักงานที่เกี่ยวข้องกับ BS ของทางบริษัท A-HOST ในแง่ของทฤษฎีและขั้นตอนต่างๆ ในการดำเนินงาน Project จะต้องทำการศึกษาระบบงานและศึกษารายงานตาม User Requirement และสร้าง Function ตามที่ได้ออกแบบไว้ ทำการจัดรูปแบบของ UI ให้สามารถแสดงได้ตามจุดประสงค์เพื่ออำนวยความสะดวกแก่ผู้ใช้งาน โดยมีการสร้างและแก้ไขปรับปรุง UI และ Code เพื่อให้ระบบ Payroll ออกมาตาม Functional Specification และทำการ POC เพื่อนำเทคโนโลยีใหม่ๆ มาใช้ใน Apex และอาจมีการปรับเปลี่ยนไปตาม User Requirement ตามลำดับขั้นตอน

สามารถแบ่งการทำงานที่ได้มีส่วนร่วม ออกเป็น 3 ขั้นตอนได้ดังนี้

- 4.1.1 พัฒนาระบบเงินเดือนด้วย Oracle Apex
- 4.1.2 POC เทคโนโลยีใหม่ๆ มาใช้ใน Oracle Apex
- 4.1.3 ทดสอบและแก้ไขข้อผิดพลาดของระบบ

4.1.1 พัฒนาระบบเงินเดือนด้วย Oracle Apex

4.1.1.1 Develop หน้าแก้ไขข้อมูลพนักงาน

หลังจากที่ได้มีการพูดคุยและประชุมหารือหน้าข้อมูลพนักงานได้มีความเห็นว่าหน้าข้อมูลพนักงานควรสามารถคำนวณหาค่าวันทดลองสิ้นสุดงานได้ทันทีทั้งนี้ทั้งนั้นได้มีการเพิ่มเติมส่วนของวันสิ้นสุดทดลองงานโดยรับค่าจากวันที่เริ่มงานซึ่งสามารถเลือกได้ 2 กรณี

1.ผ่านงาน 3 เดือน

2.ผ่านงาน 4 เดือน

โดยจะนำค่ามาคำนวณเพื่อหาวันสิ้นสุดทดลองงาน

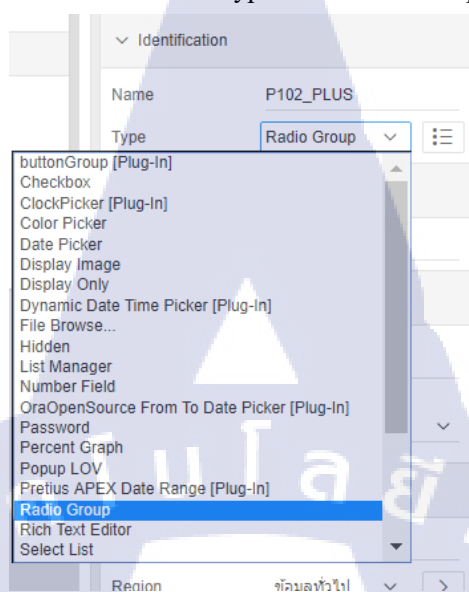
- ให้เข้ามาหน้า Edit Page ของหน้าข้อมูลพนักงาน เพื่อเพิ่มช่องทางเลือกวันทดลองสิ้นสุดงาน

รูปที่ 4.1 ขั้นตอนการเข้าหน้า Edit Page

- ให้ทำการเพิ่มไอเทมตัวเลือกสำหรับคำนวณวันสิ้นสุดทดลองงาน

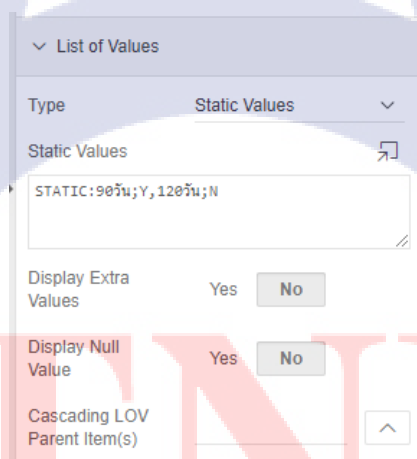
รูปที่ 4.2 ขั้นตอนการสร้างRadio Group(1)

- สร้างไอเทมให้เลือก Type เป็น Radio Group



รูปที่ 4.3 ขั้นตอนการสร้างRadio Group(2)

- เลื่อนลงมาหัวข้อ List of Value



รูปที่ 4.4 ขั้นตอนการสร้างRadio Group(3)

- เพิ่มไอเทมขึ้นมาใหม่สำหรับรับค่าจาก P102_PLUS

Page Items			+
Region	ข้อมูลทั่วไป		
20	P102_ROWID	Hidden	
30	P102_EMPL_SEQE	Hidden	
39	P102_NAME_DATE	วันที่มีผล (Effecti...	Date Picker

รูปที่ 4.5 ขั้นตอนการสร้างไอเทมรับค่าตัวแปร Radio Group (1)

- จากนั้น Set Property

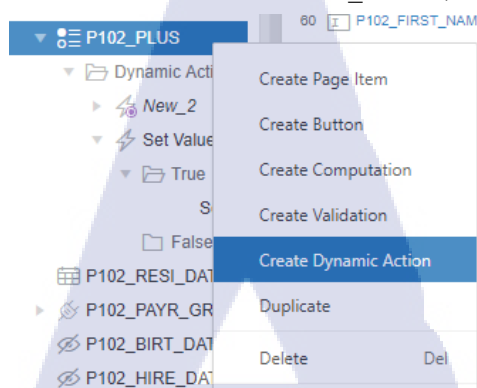
Name : ชื่อไอเทมสำหรับรับค่า

Type : Hidden

Identification	
Name	P102_NEW
Type	Hidden
Settings	
Value Protected	Yes No
Layout	
Sequence	999
Region	..ข้อมูลทั่วไป
Appearance	
Format Mask	
Advanced	
Warn on Unsaved Changes	Page Default
Source	
Type	Null
Used	Only when current val...

รูปที่ 4.6 ขั้นตอนการสร้างไอเทมรับค่าตัวแปร Radio Group (2)

- จากนั้นให้คลิกขวาไอเทมที่ P102_PLUS (Radio Group)



รูปที่ 4.7 ขั้นตอนการสร้างDynamic Action ในRadio Group(1)

- กำหนดชื่อให้กับDynamic Action

Event : Change
 Selection Type : Item(s)
 Item : P102_PLUS

Name	Set Value to P102_NEW
Execution Options	
Sequence	100
When	
Event	Change
Selection Type	Item(s)
Item(s)	P102_PLUS

รูปที่ 4.8 ขั้นตอนการสร้างDynamic Action ในRadio Group(2)

- สร้างTrue Action และกำหนด Action ดังนี้

Action : Set Value

Set type : JavaScript Expression

JS code : apex.item("P102_PLUS").getValue()

Affected Elements

Selection Type : Item(s)

Item(s) : P102_NEW

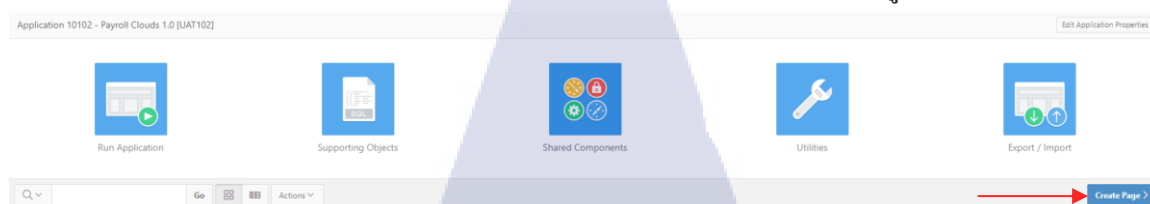
รูปที่ 4.9 ขั้นตอนการสร้างDynamic Action ในRadio Group(3)

4.1.1.1.2 Develop หน้ากำหนดชื่อ รหัส ผู้ใช้งาน

ได้ออกแบบ ฟอร์มที่แต่ละบริษัทมีผู้ดูแล Payroll Admin หลายคน

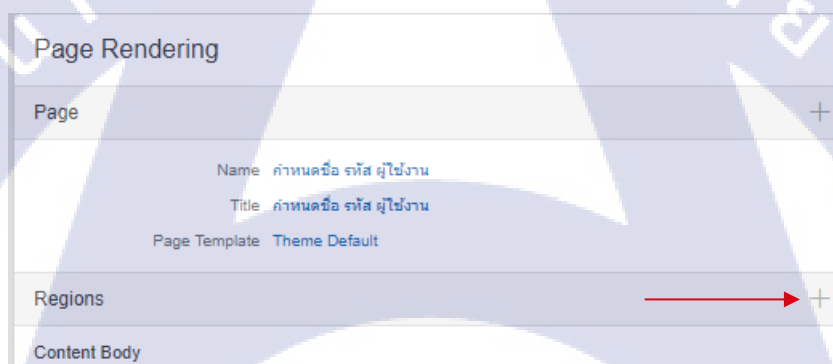
สามารถ Assign กลุ่ม Payroll ให้กับผู้ดูแลให้สามารถกำหนดสิทธิ์ในการดูงวด
ประมวลผล และได้มีการแกะ Spec เดิมจากโปรแกรม Payrollเก่า จากนั้นนำมาดี
ไซน์ให้เหมาะสมกับ Apex โดยเลือกใช้ Master Detail

- สร้างหน้าใหม่ขึ้นมา เพื่อเป็นหน้ากำหนดชื่อ รหัส ผู้ใช้งาน



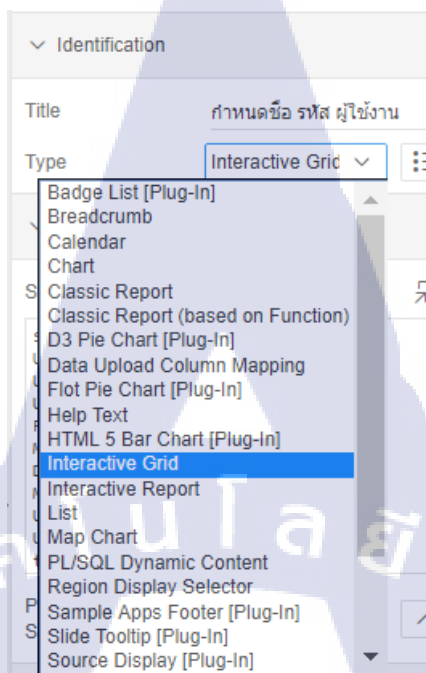
รูปที่ 4.10 ขั้นตอนการสร้างPage

- เมื่อสร้างหน้าใหม่เสร็จแล้วให้สร้าง Interactive Grid ขึ้นมา 3 ตาราง
ตารางกำหนดชื่อรหัสผู้ใช้งาน
ตารางข้อมูล
ตารางผลการประมวลผล



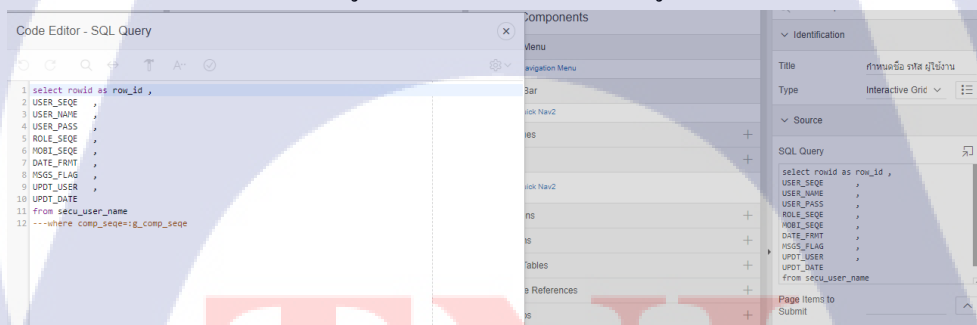
รูปที่ 4.11 ขั้นตอนการสร้าง Interactive Grid(1)

- ในส่วนของ Type เลือกเป็น Interactive Grid



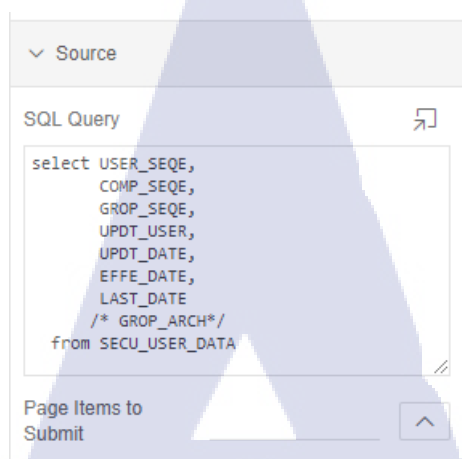
รูปที่ 4.12 ขั้นตอนการสร้าง Interactive Grid(2)

- Query ข้อมูลตาราง กำหนด ชื่อ รหัส ผู้ใช้งาน



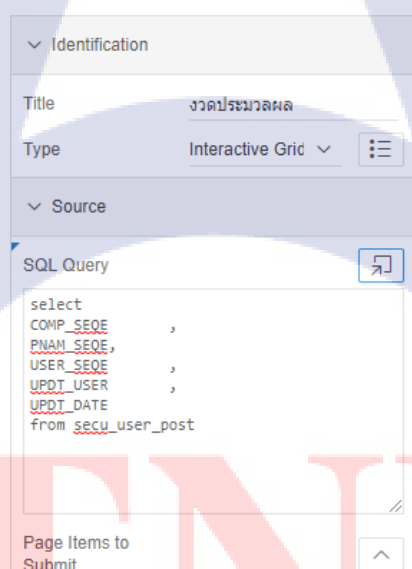
รูปที่ 4.13 ขั้นตอนการQueryของตารางกำหนด รหัส ชื่อ ผู้ใช้งาน

- สร้าง Interactive Grid และQuery เรียกข้อมูลสำหรับตารางข้อมูล



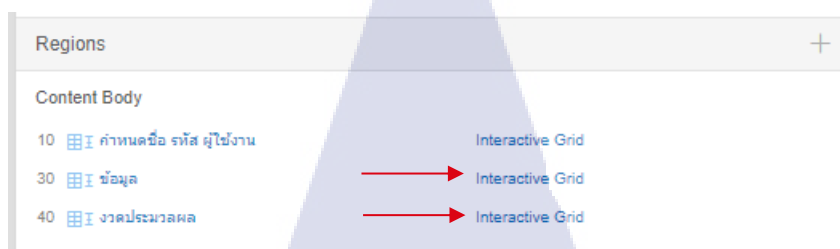
รูปที่ 4.14 ขั้นตอนการQueryของตารางข้อมูล

- สร้าง Interactive Grid และQuery เรียกข้อมูลสำหรับตารางผลการประมวลผล



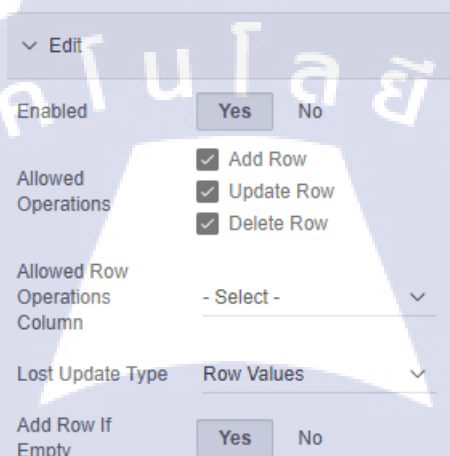
รูปที่ 4.15 ขั้นตอนการQueryของตารางผลการประมวลผล

- เมื่อเขียน SQL Query ข้อมูลแต่ละตารางแล้ว ให้ทำการ คลิก Interactive Grid เพื่อให้สามารถ Edit ข้อมูลได้อัตโนมัติ



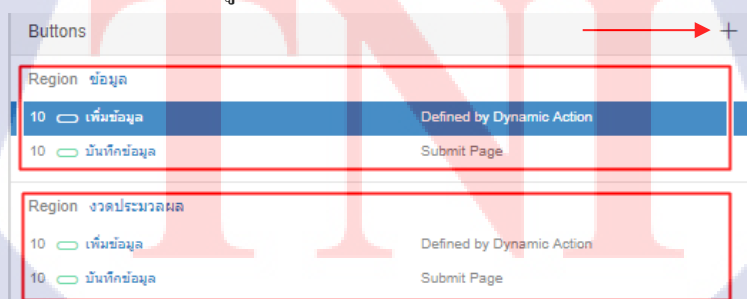
รูปที่ 4.16 ขั้นตอนการอนุญาต Edit ข้อมูล(1)

- เมื่อคลิกแล้ว ในหัวข้อ Edit ให้เลือก Enabled ปรับเป็น Yes



รูปที่ 4.17 ขั้นตอนการอนุญาต Edit ข้อมูล(2)

- จากนั้นให้ทำการสร้างปุ่ม Button สำหรับเพิ่มข้อมูลและบันทึกข้อมูลขึ้นมาในตารางข้อมูลและตารางผลการประมวลผล



รูปที่ 4.18 ขั้นตอนการสร้างปุ่มเพิ่มข้อมูลและบันทึก

- เมื่อสร้างปุ่มเสร็จแล้วให้ทำการ Dynamic Action เพื่อให้สามารถเพิ่มข้อมูลได้หลายแถวที่ BT_ADD2

Name : ตั้งชื่อให้กับ Dynamic Action

Event : Click

Selection Type : Button

Button : BT_ADD2

Action ให้เลือกExecute Javascriptsในช่องCodeให้ใส่โค้ด JavaScript

Identification

Action: Execute JavaScript

Settings

Code

```
apex.region("DATA").widget().interactiveGrid("getActions").invoke("row-add-row");
```

Affected Elements

รูปที่ 4.19 การเพิ่มแถวโดยกำหนดJavaScript

- ให้กลับไปคลิก Interactive Grid เพื่อใส่ Static Id ในการอ้างอิงตาราง เพื่อเพิ่มแถวให้กับตารางข้อมูล

Regions

Content Body

10	กำหนดชื่อ รหัส ผู้ใช้งาน	Interactive Grid
30	ข้อมูล	Interactive Grid
40	รวมประมวลผล	Interactive Grid

รูปที่ 4.20 ขั้นตอนการใส่ Static Id(1)

- หัวข้อ Advance ในช่อง Static Id ให้กำหนด ID ขึ้นมาโดยต้องไม่ซ้ำกันในแต่ละตาราง ในที่นี้ตั้งชื่อเป็น DATA

รูปที่ 4.21 ขั้นตอนการใส่ Static Id(2)

- เลื่อนลงมาที่หัวข้อ Master Detail ปรับให้ตารางข้อมูล และ งวดประมวลผล เป็น foreign key ของตารางกำหนด ชื่อ รหัส ผู้ใช้

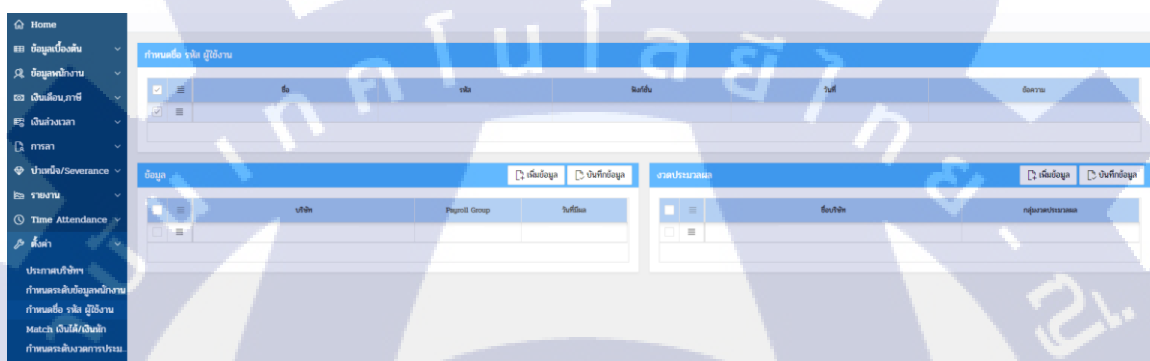
รูปที่ 4.22 ขั้นตอนกำหนด Master Detail

- เมื่อสร้าง Dynamic Action ของปุ่มเพิ่มข้อมูลของตารางข้อมูลและตารางงวดการประมวลผลแล้ว ให้ผูกปุ่มบันทึกข้อมูลกับ Process เช่น Process ของตารางข้อมูล และ งวดการประมวลผล

รูปที่ 4.23 ขั้นตอนผูกปุ่มให้กับ Process(1)

- เลือกปุ่มบันทึกข้อมูลที่สร้างไว้ในแต่ละตาราง

รูปที่ 4.24 ขั้นตอนผูกปุ่มให้กับ Process(2)



รูปที่ 4.25 UI ของหน้ากำหนดชื่อรหัส ผู้ใช้งาน

4.1.1.1.3 Develop หน้า Match เงินได้/เงินหัก

ช่วยออกแบบ หน้า Match เงินได้/เงินหัก ที่จะนำไปคำนวณต่างๆ โดยกรณีนี้องค์กรนั้นมีรหัสเงินเดือนไม่ตรงตามที่กำหนดไว้ ใช้งานสามารถกำหนดรหัสเงินได้ว่าตรงกับรหัสใดในระบบเพื่อความถูกต้องในการประมวลผลเงินเดือน และภาษี

- เมื่อทำการสร้าง Blank Page มาแล้วให้สร้าง Static Content ไว้เป็น Header ของ Interactive Grid สำหรับเรียกดูข้อมูล Match เงินได้/เงินหัก

Page	+
Name	Match เงินได้/เงินหัก
Title	Match เงินได้/เงินหัก
Page Template	Theme Default
Regions	→ +

รูปที่ 4.26 ขั้นตอนการสร้าง Static Content(1)

- กำหนดชื่อ Title เป็น Match เงินได้/เงินหัก

Identification	
Title	Match เงินได้/เงินหัก
Type	Static Content
Source	
Text	
Layout	
Sequence	10
Parent Region	- Select -
Position	Content Body
Start New Row	Yes No
Column	Automatic
Column Span	Automatic
Column CSS Classes	
Column Attributes	
Appearance	
Template	Standard
Template Options	Use Template Defaults, Scroll - Default
CSS Classes	

รูปที่ 4.27 ขั้นตอนการสร้าง Static Content(2)

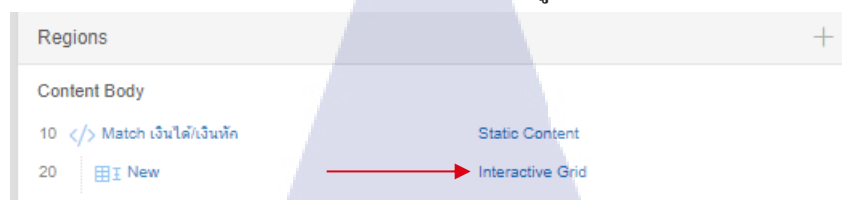
- คลิกปุ่ม + ในหัวข้อ Regions เพื่อสร้าง Interactive Grid

รูปที่ 4.28 ขั้นตอนการสร้าง Interactive Grid Match เงินได้/เงินหัก(1)

- Type : Interactive Grid
- SQL Query : select * from desc_incm_tabl where
Comp_seqe = :g_comp_seqe
- Parent Region : Match เงินได้/เงินหัก

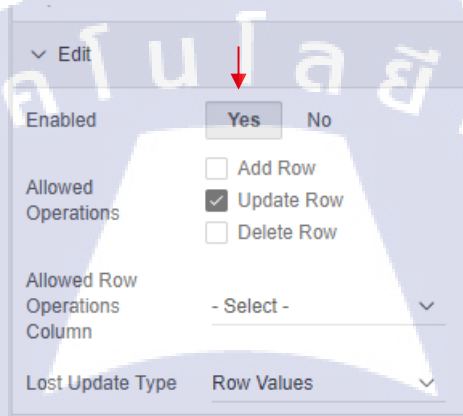
รูปที่ 4.29 ขั้นตอนการสร้าง Interactive Grid Match เงินได้/เงินหัก(2)

- กำหนด Enable สามารถแก้ไขข้อมูลได้ โดยคลิกที่ Interactive Grid



รูปที่ 4.30 ขั้นตอน Enable Interactive Grid Match เงินได้/เงินหัก(1)

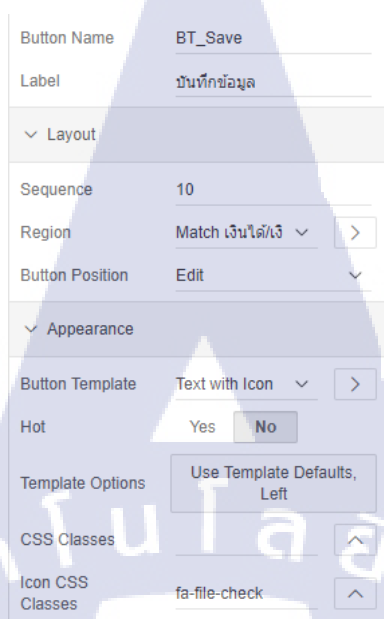
- Enabled ให้เลือกเป็น Yes จากนั้นคลิกออก Add row และ Delete row



รูปที่ 4.31 ขั้นตอน Enable Interactive Grid Match เงินได้/เงินหัก(2)

- สร้างปุ่มสำหรับการบันทึกแก้ไขข้อมูล กำหนดชื่อปุ่มและกำหนดให้ปุ่มอยู่ใน Region Match เงินได้/เงินหัก , Button Position ให้เลือกเป็น Edit

TNI



Button Name	BT_Save
Label	บันทึกข้อมูล
Layout	
Sequence	10
Region	Match เงินได้/เงินหัก
Button Position	Edit
Appearance	
Button Template	Text with Icon
Hot	Yes ☐ No ☒
Template Options	Use Template Defaults, Left
CSS Classes	
Icon CSS Classes	fa-file-check

รูปที่ 4.32 ขั้นตอนการสร้างปุ่มสำหรับบันทึกข้อมูล Match เงินได้/เงินหัก

- สร้าง Process เพื่อเขียนสคริปสำหรับแก้ไขข้อมูล

Type : PL/SQL CODE

PL/SQL CODE : begin

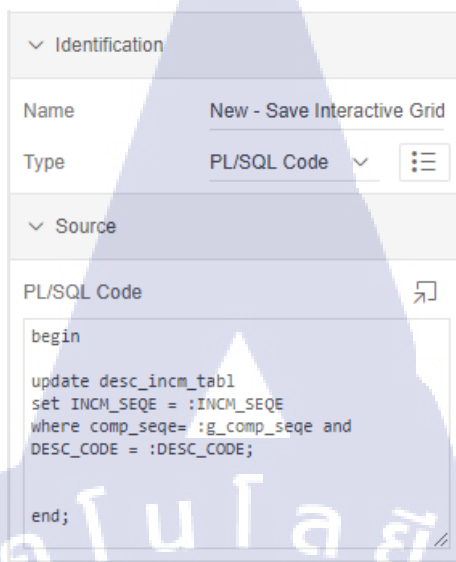
Update desc_incm_tabl set

Incm_seqe = : incm_seqe

Where comp_seqe = :g_comp_seqe

And desc_code = :desc_code;

End;



▼ Identification

Name New - Save Interactive Grid

Type PL/SQL Code ▼

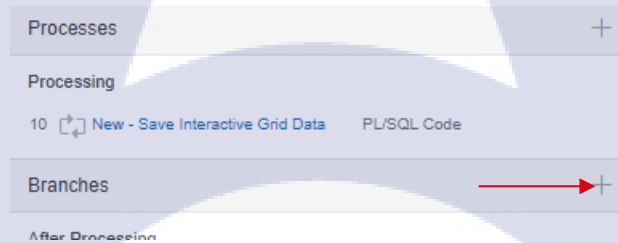
▼ Source

PL/SQL Code

```
begin
update desc_incmtab1
set INCM_SEQE = :INCM_SEQE
where comp_seqe= :g_comp_seqe and
DESC_CODE = :DESC_CODE;
end;
```

รูปที่ 4.33 ขั้นตอนการสร้าง Process Match เงินได้/เงินหัก

- ไปหัวข้อ Branches เพื่อเป็นการกำหนดว่า เมื่อเพิ่มข้อมูลเสร็จให้ลิงค์ไปหน้าใด



Processes +

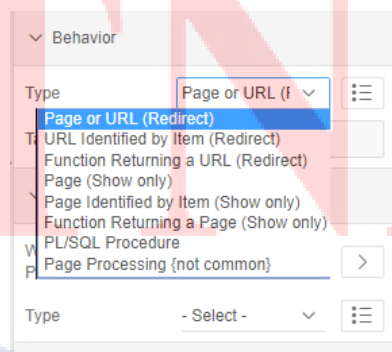
Processing

10 New - Save Interactive Grid Data PL/SQL Code

Branches →

รูปที่ 4.34 ขั้นตอนลิงค์ไปยังหน้าที่ต้องการ(1)

- เลือก Type เป็น Page or URL(Redirect)



▼ Behavior

Type Page or URL (f) ▼

- Page or URL (Redirect)
- URL Identified by Item (Redirect)
- Function Returning a URL (Redirect)
- Page (Show only)
- Page Identified by Item (Show only)
- Function Returning a Page (Show only)
- PL/SQL Procedure
- Page Processing (not common)

Type - Select - ▼

รูปที่ 4.35 ขั้นตอนลิงค์ไปยังหน้าที่ต้องการ(2)

- หัวข้อBehavior ตัวเลือก Target ให้เลือก Type กำหนดหน้าที่ต้องการลิงค์ สามารถตั้งเคลียร์เซสชันหน้าได้ตามต้องการ

รูปที่ 4.36 ขั้นตอนลิงค์ไปยังหน้าที่ต้องการ(3)

เงิน	เงินภาษี	เงินภาษีหัก	เงินได้ / เงินหัก
SAL	เงินเดือน	Salary	BA31 : เงินเดือน
TAX	ภาษี	Tax	TAX : ภาษีเงินได้
SSEE	ประกันสังคม	Social Security	SSEE : ประกันสังคมส่วนผู้จ้าง
SSER	ประกันสังคมส่วน	Social Security ER	SSER : ประกันสังคมส่วนนาย
PFEE	กองทุนสำรองเลี้ยงชีพ	Provident Fund EE	PFEE : กองทุนสำรองเลี้ยงชีพ
PFER	กองทุนสำรองเลี้ยงชีพส่วน	Provident Fund ER	PFER : กองทุนสำรองเลี้ยงชีพ

รูปที่ 4.37 UI Match หน้าเงินได้/เงินหัก

4.1.1.1.4 Develop หน้ากำหนดระดับงวดการประมวลผล

ช่วยออกแบบหน้ากำหนดระดับงวดประมวลผล โดยหน้านี้สามารถกำหนดสิทธิ์ในการ Access งวดการประมวลผล

- สร้างInteractive Grid ขึ้นมาโดยกำหนด

Type : Interactive Grid
 SQL Query : select rowed as row_id,comp_seqe,
 Pnam_seqe , pname_name , updt_user
 ,updt_date from secu_post_name

▼ Identification

Title ระดับ

Type Interactive Grid

▼ Source

SQL Query

```
select rowid as Row_id ,COMP_SEQE
,PNAM_SEQE ,PNAM_NAME ,UPDT_USER
,UPDT_DATE from secu_post_name
```

Page Items to Submit

รูปที่ 4.38 ขั้นตอนการสร้างInteractive Grid กำหนดระดับงวดประมวลผล

- สร้างInteractive Grid ขึ้นมาโดยให้ชื่อ งวดการประมวล
 เลือกType เป็น Interactive Grid และเขียน SQL เพื่อQuery ข้อมูลมาโชว์
 SQL Query : select rowid as row_id ,comp_seqe,pname_seqe
 ,post_seqe , updt_user,updt_date from secu_post_data

TNI

รูปที่ 4.39 ขั้นตอนการสร้าง Interactive Grid งวดการประมวล

- สร้างปุ่มเพิ่มแถวและบันทึกของทั้ง2ตาราง ให้ทำการสร้าง Dynamic Action การทำงานของปุ่มขึ้นมา

Name : add row

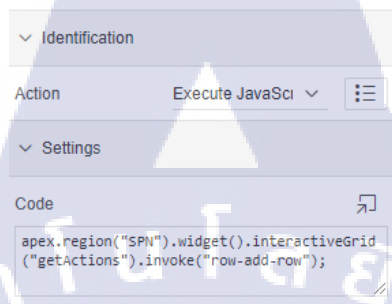
Event : Click

รูปที่ 4.40 ขั้นตอนการสร้าง Dynamic Action ปุ่มเพิ่มข้อมูลตารางระดับ(1)

- สร้างเงื่อนไข True โดยกำหนด

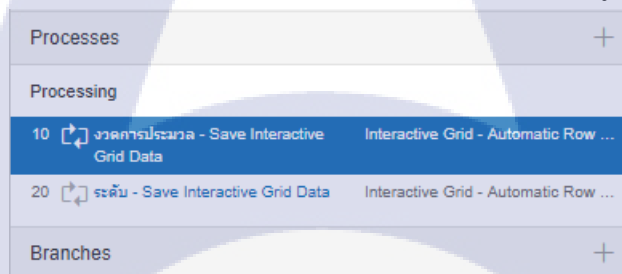
Action : Execute JavaScript

Code : apex.region("SPN").widget().
InteractiveGrid("getAction").invoke("row-add-row");



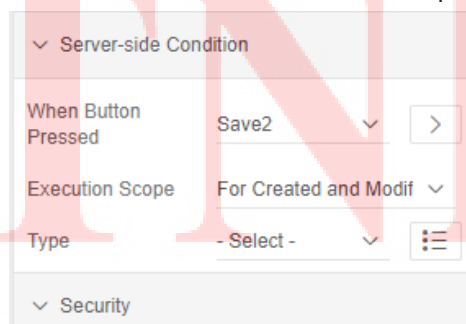
รูปที่ 4.41 ขั้นตอนการสร้าง Dynamic Actionปุ่มเพิ่มข้อมูลตารางระดับ(2)

- เมื่อทำDA การเพิ่มแถวทั้ง2ตารางแล้วให้กลับมาผูกปุ่มButtonกับProcess



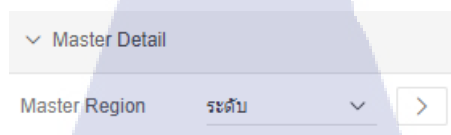
รูปที่ 4.42 ขั้นตอนการผูกปุ่มButtonกับProcessจาดการประมวล(1)

- ในหัวข้อ When Button Pressed ให้เลือกปุ่มที่ทำการสร้างไว้สำหรับตาราง



รูปที่ 4.43 ขั้นตอนการผูกปุ่มButtonกับProcessจาดการประมวล(2)

- คลิกที่งวดการประมวลเพื่อ Set Master Detail เลื่อนลงมาหัวข้อ Master Detail เลือก ตารางระดับ



รูปที่ 4.44 ขั้นตอนการกำหนด Master Detail งวดการประมวล



รูปที่ 4.45 UI ของกำหนดระดับงวดการประมวล

4.1.1.1.5 Develop หน้าเงื่อนไข , สิทธิประโยชน์

ปรับปรุงในส่วนกรอกข้อมูลของค่าลดหย่อนภาษีโดย ค่าลดหย่อนภาษีที่จะเพิ่มข้อมูลหรือเปลี่ยนไปทุกปีและจะมีบางข้อที่เหมือนเดิม เช่น บุตร หรือ ประกันชีวิต เพราะฉะนั้นจึงได้มีการพูดคุยระหว่างทีมในการสร้าง Tab-container ค่าลดหย่อนที่เปลี่ยนไปทุกปี กับ ค่าลดหย่อนที่เหมือนเดิมตลอด และรวมถึง การตรวจสอบค่าขกยอดสะสม

- สร้าง Interactive Grid
กำหนด ชื่อ Title : ลดหย่อนภาษี (Tax Allowance Fix)
Type : Interactive Grid
SQL Query : select rowid as row_id ,EMPL_TAXX_ALWF.*
from EMPL_TAXX_ALWF
where empl_seqe=:p174_empl_seqe

```

and comp_seqe=:g_comp_seqe
and exists(select null from secu_grop_empl b
where comp_seqe=:g_comp_seqe
and grop_seqe=:g_grop_seqe
and empl_seqe=b.empl_seqe);

```

Parent Region : Tab

Identification

Title: ลดหย่อนภาษี (Tax Allowance Fix)

Type: Interactive Grid

Source

SQL Query

```

select rowid as row_id ,EMPL_TAXX_ALWF.*
  from EMPL_TAXX_ALWF
 where empl_seqe=:p174_empl_seqe
    and comp_seqe=:g_comp_seqe
    and exists(select null from
      secu_grop_empl b
        where
          comp_seqe=:g_comp_seqe
        and
          grop_seqe=:g_grop_seqe
        and

```

Page Items to Submit

Layout

Sequence: 32

Parent Region: .tab

รูปที่ 4.46 ขั้นตอนการสร้าง Interactive Grid ลดหย่อนภาษี (Tax Allowance Fix)

- สร้างตารางค่าสะสมขึ้นมา

Title :ค่าสะสม (YTOD)

Type :Interactive Grid

SQL Query : select rowid as row_id, empl_empl_ytod.*

From empl_empl_ytod

Where comp_seqe=:g_comp_seqe and

empl_seqe=:P174_empl_seqe;

The screenshot shows the configuration for an Interactive Grid titled 'ค่าสะสม (YTOD)'. The 'Source' section contains an SQL query: `select rowid as row_id ,empl_empl_ytod.* from empl_empl_ytod where comp_seq=:g_comp_seqe and empl_seqe = :P174_EMPL_SEQE;`. The 'Layout' section shows 'Sequence' set to 40 and 'Parent Region' set to '..tab'.

รูปที่ 4.47 ขั้นตอนการสร้าง Interactive Grid ค่าสะสม (YTOD)

- สร้างปุ่มสำหรับเพิ่มข้อมูลและบันทึกข้อมูลของแต่ละตาราง

The screenshot displays three regions on a page:

- Region: ลดหย่อนภาษี (Tax Allowance Var)**
 - Item 10: Button 'เพิ่มข้อมูล' (Add Data) with a green icon.
 - Item 20: Button 'บันทึกข้อมูล' (Save Data) with a green icon.
 - Item 20: Button 'เพิ่มข้อมูล' (Add Data) with a green icon.
 - Dynamic Actions: 'Go To Page 156' and 'Go To Page 193'.
- Region: ลดหย่อนภาษี (Tax Allowance Fix)**
 - Item 10: Button 'เพิ่มข้อมูล' (Add Data) with a green icon.
 - Item 20: Button 'บันทึกข้อมูล' (Save Data) with a green icon.
 - Dynamic Actions: 'Defined by Dynamic Action' and 'Submit Page'.
- Region: ค่าสะสม (YTOD)**
 - Item 10: Button 'เพิ่มข้อมูล' (Add Data) with a green icon.
 - Item 20: Button 'บันทึกข้อมูล' (Save Data) with a green icon.
 - Dynamic Actions: 'Defined by Dynamic Action' and 'Submit Page'.

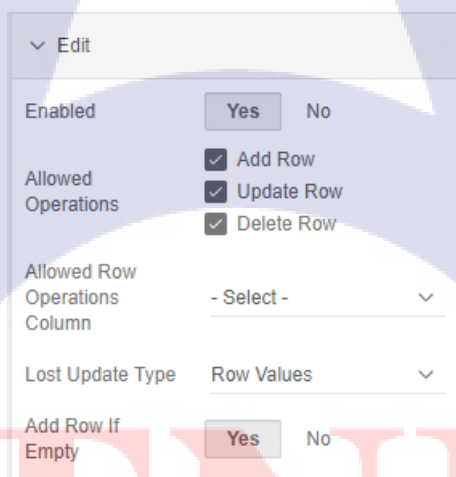
รูปที่ 4.48 ปุ่มเพิ่มข้อมูลและบันทึกข้อมูล

- คลิก Interactive Grid ของตารางค่าสะสม (YTOD) ตารางลดหย่อนภาษี (Tax Allowance Fix) เพื่อให้สามารถเพิ่ม แก้ไข ลบ ของข้อมูลได้



รูปที่ 4.49 ขั้นตอนการ Enable เพิ่มแก้ไขแถวข้อมูล(1)

- Enable ปรับเป็น Yes



รูปที่ 4.50 ขั้นตอนการ Enable เพิ่มแก้ไขแถวข้อมูล(2)

4.1.1.1.6 Develop หน้าบันทึกเงินล่วงหน้า

เนื่องจากได้รับ Requirement จากพี่พนักงานที่ดูแล แผนกBS มีความต้องการให้ระบบPayroll สามารถโชว์ผลลัพธ์ที่แสดง ปิดงวด หรือเปิดงวด หากงวดประมวลผลงวดใดถูกปิดงวดไปแล้ว จะแสดงข้อความว่า Y หากงวดใดที่ยังไม่ทำการปิดงวด ให้แสดงค่าเปล่าไว้ โดยงวดที่ถูกปิดแล้วจะไม่สามารถ แก้ไข เพิ่มข้อมูล หรือ ลบข้อมูล ได้อีก

- สร้างไอเทมสำหรับโชว์ค่าปิดงวด เปิดงวด โดยกำหนด

Name	: P138_TF_PROC_FLAG
Type	: Display Only
Label	: ปิดงวด (Closing Period)
Region	: บันทึกเงินล่วงหน้า

TNI

▼ Identification

Name P138_TF_PROC_FLAG

Type Display Only

▼ Label

Label ปิดงวด (Closing Period)

▼ Settings

Save Session State ☒ Yes ☐ No

Based On Page Item Value

Show Line Breaks ☒ Yes ☐ No

▼ Layout

Sequence 20

Region บันทึการเงินล่วงหน้า

Start New Row ☒ Yes ☐ No

Column 4

Column Span Automatic

Label Column Span Page Template Default

Column CSS Classes

Column Attributes

รูปที่ 4.51 ขั้นตอนการสร้างไอเทมไอวอร์ค Flag

- สร้าง Dynamic Action สำหรับ Disable ปุ่ม

กำหนดชื่อให้กับ Dynamic Action

Event ให้เลือกเป็น Change

ในหัวข้อ Client – Side Condition กำหนดให้ Item is not null

Item ให้เลือกเป็นไอเทมที่ไว้สำหรับค่า Flag

▼ Identification

Name disable when closing period

▼ Execution Options

Sequence 110

▼ When

Event Change ▼

Selection Type Item(s) ▼

Item(s) P138_TF_PROC_F ^

▼ Client-side Condition

Type Item is not null ▼

Item P138_TF_PROC_F ^

รูปที่ 4.52 ขั้นตอนการสร้างDynamic Action Disable Button(1)

- สร้าง True Action ขึ้นมา
Action เลือก Disable
Selection Type เลือก Button
Button ให้เลือกปุ่มที่ต้องการปิด

▼ Identification

Action Disable ▼

▼ Affected Elements

Selection Type Button ▼

Button ADDROW ^

▼ Execution Options

Sequence 10

Event disable when clo ^

Fire When Event Result Is True ▼

Fire on Initialization Yes No

รูปที่ 4.53 ขั้นตอนการสร้างDynamic Action Disable Button(2)

- สร้างเงื่อนไข False

Action ให้เลือกเป็น Enable หากไม่เป็นจริงให้โชว์ปุ่มขึ้นมา

Selection Type เลือกเป็น Button

The screenshot shows the configuration for a Dynamic Action. Under the 'Identification' section, the 'Action' is set to 'Enable'. In the 'Affected Elements' section, the 'Selection Type' is 'Button' and the specific button is 'ADDROW'. Under 'Execution Options', the 'Sequence' is 10, the 'Event' is 'disable when clo', 'Fire When Event Result Is' is 'False', and 'Fire on Initialization' has 'Yes' selected.

รูปที่ 4.54 ขั้นตอนการสร้างDynamic Action Disable Button(3)

- จากนั้นให้คลิกขวาเพื่อสร้าง Dynamic Action ที่ไอเทมชื่อ P138_PL_INCM_POST_CODE ที่ไว้สำหรับเลือกงวดประมวลผลต่างๆ กำหนดชื่อ และ Event เลือกเป็น Change

The screenshot shows the configuration for a Dynamic Action. Under 'Identification', the 'Name' is 'change pl then show Y/N'. Under 'Execution Options', the 'Sequence' is 30. In the 'When' section, the 'Event' is 'Change', the 'Selection Type' is 'Item(s)', and the 'Item(s)' is 'P138_PL_INCM_PC'. The 'Client-side Condition' section is visible at the bottom.

รูปที่ 4.55 ขั้นตอนการสร้างDynamic Action Show flag(1)

- คลิกขวาเพื่อสร้าง True Action

Action : Execute PL/SQL

PL/SQL CODE :

```
select POST_FLAG into :P138_TF_PROC_FLAG from incm_post_para
```

```
where comp_seqe=:g_comp_seqe
```

```
and post_seqe=:P138_PL_INCM_POST_CODE ;
```

```
exception WHEN NO_DATA_FOUND
```

```
THEN :P138_TF_PROC_FLAG :=null;
```

Item to submit : P138_PL_INCM_POST_CODE

Item to Return : P138_TF_PROC_FLAG

▼ Identification

Action Execute PL/SC ⋮

▼ Settings

PL/SQL Code 📄

```
select POST_FLAG into :P138_TF_PROC_FLAG
from incm_post_para
where comp_seqe=:g_comp_seqe
and post_seqe=:P138_PL_INCM_POST_CODE ;
exception WHEN NO_DATA_FOUND THEN
:P138_TF_PROC_FLAG := null;
```

Items to Submit P138_PL_INCM_PC ⬆

Items to Return P138_TF_PROC_F ⬆

Suppress Change Event Yes No

▼ Execution Options

Sequence 20

Event change pl then >

Fire When Event Result Is True ▼

Fire on Initialization Yes No

Stop Execution On Error Yes No

Wait For Result Yes No

รูปที่ 4.56 ขั้นตอนการสร้าง Dynamic Action Show flag(2)

พนักงาน	วันที่ทำ OT	ประเภทเงินล่วงเวลา	เวลาพัก	เวลาเริ่ม	จำนวนชั่วโมง	จำนวนเงิน	หมายเหตุ
นายสมชาย งามใจ	25-08-2018	1ST	19:00	19:00	1.00	312.50	-
นายสมชาย งามใจ	25-08-2018	1ST	19:00	20:00	1.00	281.25	-

รูปที่ 4.57 UI ของหน้าบันทึกเงินล่วงเวลา

4.1.1.1.7 Develop หน้าประเภทการลา (Define Leave Type)

ออกแบบหน้าประเภทการลา หน้านี้แสดงรหัสการลา ของการลาแต่ละประเภท และสามารถเพิ่ม แก้ไข ข้อมูลของประเภทการลาได้

- สร้าง Interactive Grid สำหรับกำหนดประเภทการลา

SQL code : select ROWID as ROW_ID,
comp_seqe,
LEAV_SEQE, LEAV_CODE,
LEAV_TNAM, LEAV_ENAM,
SYST_CODE, EFFE_DATE,
apex_item.checkbox (1, ENAB_FLAG,
case when ENAB_FLAG = 'Y'
then 'CHECKED ' else " end || ' DISABLED') as
ENAB_FLAG, updt_user, updt_date
from TABL_LEAV_CODE
where comp_seqe=:g_comp_seqe;

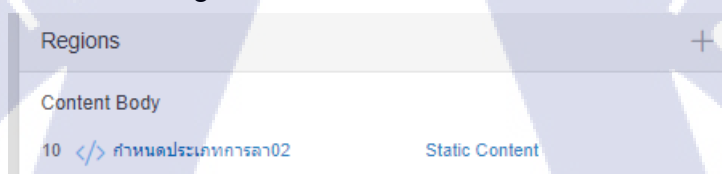
```

1 select ROWID as ROW_ID,
2      comp_seqe,
3      LEAV_SEQE,
4      LEAV_CODE,
5      LEAV_TNAM,
6      LEAV_ENAM,
7      SYST_CODE,
8      EFFE_DATE,
9      apex_item.checkbox ( 1, ENAB_FLAG,
10     case when ENAB_FLAG = 'Y'
11     then 'CHECKED' else '' end || ' DISABLED') as ENAB_FLAG,|
12      updt_user,
13      updt_date
14 from TABL_LEAV_CODE
15 where comp_seqe=:g_comp_seqe;

```

รูปที่ 4.58 ขั้นตอนการสร้างInteractive Grid กำหนดประเภทการลา

- สร้างหน้าใหม่เพื่อเป็นหน้าสำหรับแบบฟอร์มเพิ่มข้อมูลรหัสการลา
- สร้างRegion Static Content ขึ้นมาเพื่อเป็นหัวข้อHeader



รูปที่ 4.59 ขั้นตอนการแบบฟอร์มเพิ่มข้อมูลประเภทการลา

-
- สร้างItem สำหรับกรอกข้อมูล

P185_ROWID	Type	:Hidden
P185_LEAV_SEQE	Type	:Hidden
Source		
Type	: Database Column	
Database Column	: LEAV_SEQE	
▼ Source		
Type	Database Colu	⋮
Database Column	LEAV_SEQE	
Used	Always, replacing any ▼	
Post Calculation Computation		

รูปที่ 4.60 ขั้นตอนการแบบฟอร์มเพิ่มข้อมูลประเภทการลา

- P185_LEAV_CODE Type :Text Field
- P185_LEAV_TNAM Type :Text Field
- P185_LEAV_ENAM Type :Text Field
- P185_SYST_CODE Type :Select list
- List of Value : SQL Query
 select leav_tnam d , syst_code r from
 tabl_leav_code
 where comp_seqe=:g_comp_seqe
- P185_EFFE_DATE Type :Date picker
- P185_ENAB_FLAG Type :Checkbox
- List of Values : Static Values
 STATIC;;Y
- P185_UPDT_USER Type :Hidden
- Source : Database Column
- Database Column : UPDT_USER
- P185_UPDT_DATE Type :Hidden
- Source : Database Column
- Database Column : UPDT_DATE

- จากนั้นสร้าง Process Insert , Update , Delete

Insert Code :

```
begin
insert into TABL_LEAV_CODE(
    comp_seqe, LEAV_SEQE,
    LEAV_CODE, LEAV_TNAM,
    LEAV_ENAM, SYST_CODE,
    EFFE_DATE, ENAB_FLAG,
    updt_user, updt_date)
```

```

values
(
:g_comp_seqe,
SEQ_TABL_LEAV_CODE.nextval,
:P185_LEAV_CODE, :P185_LEAV_TNAM,
:P185_LEAV_ENAM, :P185_SYST_CODE,
:P185_EFFE_DATE, :P185_ENAB_FLAG,
:g_user_name, sysdate);
end;

Update Code :
begin
UPDATE TABL_LEAV_CODE set
    LEAV_CODE = :P185_LEAV_CODE,
    LEAV_TNAM = :P185_LEAV_TNAM,
    LEAV_ENAM = :P185_LEAV_ENAM,
    SYST_CODE = :P185_SYST_CODE,
    EFFE_DATE = :P185_EFFE_DATE,
    ENAB_FLAG = :P185_ENAB_FLAG,
    updt_user = :P185_UPDT_USER,
    updt_date = sysdate
where rowid = :P185_ROWID; end;

Delete Code :
begin
delete from TABL_LEAV_CODE
where rowid = :P185_ROWID;
end;

```




Processing	
10	Insert PL/SQL Code
20	update PL/SQL Code
30	delete PL/SQL Code

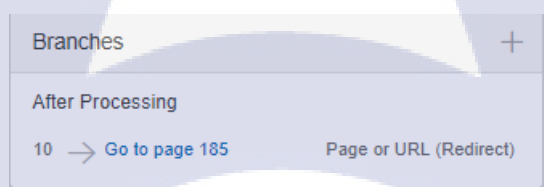
รูปที่ 4.61 ขั้นตอนการเขียนplsql insert , update , delete

- กลับไปหน้ากำหนดประเภทการลา สร้างปุ่มเพิ่มข้อมูลขึ้นมา



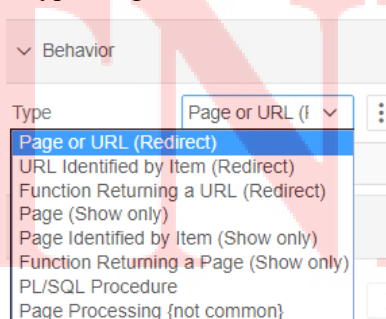
รูปที่ 4.62 ขั้นตอนการสร้างปุ่มเพิ่มข้อมูลกำหนดประเภทการลา

- กำหนดลิงก์หน้าไปยังหน้าแบบฟอร์ม



รูปที่ 4.63 ขั้นตอนการกำหนดBranch(1)

- กำหนด Type : Page Or URL (Redirect)



รูปที่ 4.64 ขั้นตอนการกำหนดBranch(2)

- Type : Page In This Application
Page : ใส่เลขหน้าแบบฟอร์มลงไป

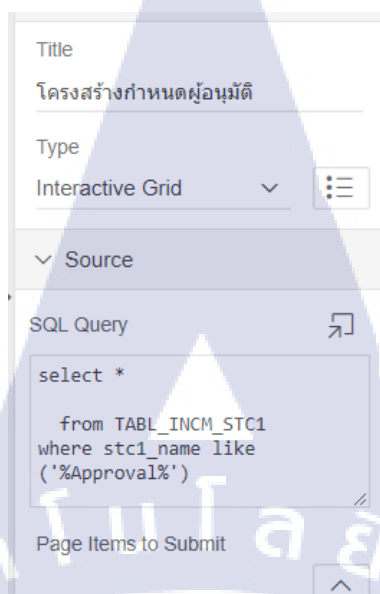
รูปที่ 4.65 ขั้นตอนการกำหนดBranch(3)

4.1.1.1.8 Develop หน้าโครงสร้างกำหนดผู้อนุมัติ

ได้ออกแบบตามRequirementให้ตารางโครงสร้างกำหนดผู้อนุมัติส่วนรายละเอียด สามารถแสดงลิสต์รายชื่อของพนักงานได้ และส่งค่ากลับเป็นรหัสพนักงาน โดยไม่ต้องป้อนรหัสพนักงานเอง

- สร้าง Interactive Grid ขึ้นมา2ตาราง คือ ตารางโครงสร้างกำหนดผู้อนุมัติและตารางโครงสร้างกำหนดผู้อนุมัติส่วนรายละเอียด

Title : โครงสร้างกำหนดผู้อนุมัติ
Type : Interactive Grid
SQL Query : select *
from TABL_INCM_STC1
where stc1_name like ('%Approval%')



Title
โครงสร้างกำหนดผู้อนุมัติ

Type
Interactive Grid

Source
SQL Query

```
select *
from TABL_INCM_STC1
where stc1_name like
('%Approval%')
```

Page Items to Submit

รูปที่ 4.66 ขั้นตอนการสร้าง Interactive Grid ตารางโครงสร้างกำหนดผู้อนุมัติ(1)

- สร้าง Interactive Grid ขึ้นมาโดยกำหนด

Title	: โครงสร้างกำหนดผู้อนุมัติส่วนรายละเอียด
Type	: Interactive Grid
SQL Query	: select rowid as row_id, COMP_SEQE, STC1_SEQE, STC2_SEQE, STC2_CODE, STC2_NAME, UPDT_USER, UPDT_DATE from TABL_INCM_STC2 where comp_seqe=:g_comp_seqe

The screenshot shows the configuration interface for an Interactive Grid. The 'Identification' tab is active, displaying the following fields:

- Title: โครงสร้างกำหนดผู้อนุมัติส่วนรายละเอียด
- Type: Interactive Grid

The 'Source' tab is also visible, showing the following SQL Query:

```
select rowid as
row_id,COMP_SEQE,
STC1_SEQE,
STC2_SEQE,
STC2_CODE,
STC2_NAME,
UPDT_USER,
UPDT_DATE
from TABL_INCM_STC2
where comp_seqe=:g_comp_seqe
```

รูปที่ 4.67 ขั้นตอนการสร้างInteractive Gridตารางกำหนดผู้อนุมัติส่วนรายละเอียด(1)

- กำหนด Master Detail เป็นโครงสร้างกำหนดผู้อนุมัติ Static Id เป็น STC2

The screenshot shows the configuration interface for an Interactive Grid. The 'Master Detail' tab is active, displaying the following fields:

- Master Region: โครงสร้างกำหนดผู้อนุมัติ
- Static ID: STC2
- Custom Attributes: (empty)
- Region Image: (empty)

รูปที่ 4.68 ขั้นตอนการสร้างInteractive Gridตารางกำหนดผู้อนุมัติส่วนรายละเอียด(2)

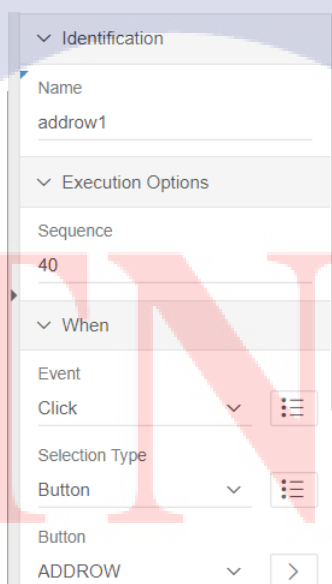
- เมื่อสร้างทั้งสองตารางเสร็จแล้วให้ทำการสร้างปุ่มเพิ่มข้อมูลและบันทึกข้อมูล ทั้งสองตาราง

Buttons		+
Region โครงสร้างกำหนดผู้อนุมัติ		
10	 เพิ่มข้อมูล	Defined by Dynamic Action
20	 บันทึก	Submit Page
Region โครงสร้างกำหนดผู้อนุมัติส่วนรายละเอียด		
10	 เพิ่มข้อมูล	Defined by Dynamic Action
20	 บันทึก	Submit Page

รูปที่ 4.69 ขั้นตอนการสร้างปุ่มเพิ่มข้อมูลและบันทึกข้อมูล

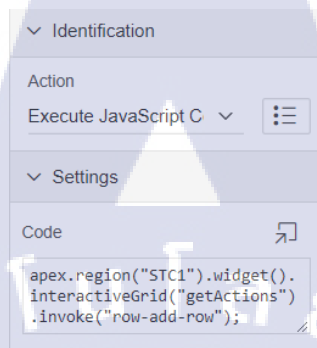
- ให้สร้าง Dynamic Action ที่ปุ่มเพิ่มข้อมูลของตาราง โครงสร้างกำหนดผู้อนุมัติ

Name :ชื่อของDynamic Action
 Event :Click
 Selection Type :Button
 Button :ADDROW



รูปที่ 4.70 ขั้นตอนการสร้างDynamic Action ปุ่มเพิ่มข้อมูลที่ตารางโครงสร้างกำหนดผู้อนุมัติ(1)

- Action :Execute JavaScript
- Code : apex.region("STC1").widget().interactiveGrid("getActions").invoke("row-add-row");



รูปที่ 4.71 ขั้นตอนการสร้างDynamic Actionปุ่มเพิ่มข้อมูลที่ตารางโครงสร้างกำหนดผู้อนุมัติ(2)

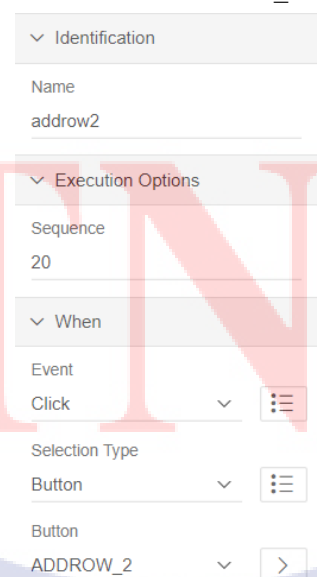
- ให้สร้าง Dynamic Action ที่ปุ่มเพิ่มข้อมูลของตารางโครงสร้างกำหนดผู้อนุมัติส่วนรายละเอียด

Name :ชื่อของDynamic Action

Event :Click

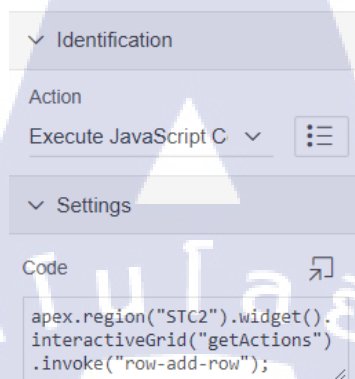
Selection Type :Button

Button :ADDROW_2



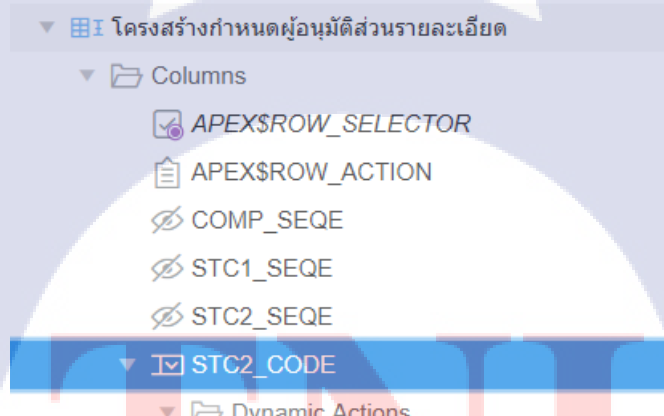
รูปที่ 4.72 ขั้นตอนการสร้างDynamic Actionปุ่มเพิ่มข้อมูลที่กำหนดผู้อนุมัติส่วนรายละเอียด(1)

- Action :Execute JavaScript
- Code : apex.region("STC2").widget().interactiveGrid("getActions").invoke("row-add-row");



รูปที่ 4.73 ขั้นตอนการสร้างDynamic Actionปุ่มเพิ่มข้อมูลที่กำหนดผู้อนุมัติส่วนรายละเอียด(2)

- ใน Column STC2_CODE ให้ปรับเป็นSelect list



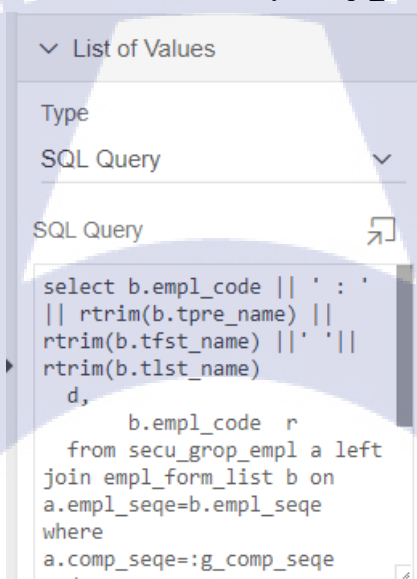
รูปที่ 4.74 ขั้นตอนการปรับคอลัมน์เป็นSelect list(1)

- Type : SQL Query
- SQL Query : select b.empl_code || ':' || rtrim(b.tpre_name) || rtrim(b.tfst_name) || ' ' || rtrim(b.tlst_name)

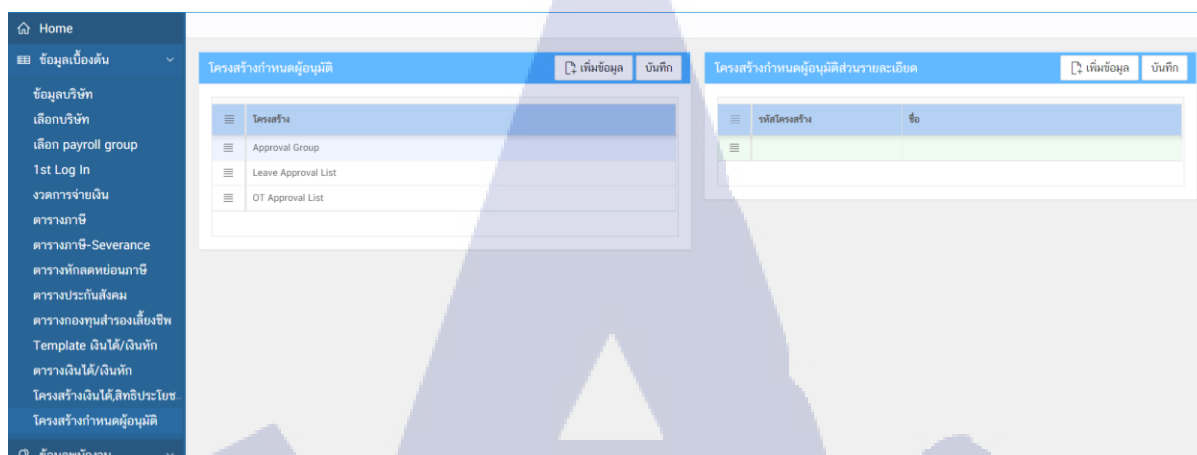
```

d, b.empl_code r
from secu_grop_empl a left join empl_form_list b on
a.empl_seqe=b.empl_seqe
where a.comp_seqe=:g_comp_seqe
and a.grop_seqe=:g_grop_seqe
and b.name_date=(select max(c.effe_date)
from empl_empl_name c
where c.empl_seqe=b.empl_seqe
and c.effe_date<=sysdate)
order by b.empl_code

```



รูปที่ 4.75 ขั้นตอนการปรับคัดมัน้เป็นSelect list(2)



รูปที่ 4.76 UI หน้าโครงสร้างกำหนดผู้อนุมัติ

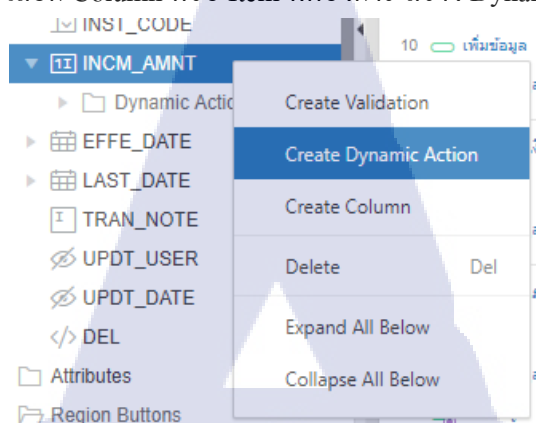
4.1.1.1.9 Dynamic Action Set Format

เนื่องจากได้รับ Requirement เมื่อข้อมูลจำนวนเงินที่ยังไม่ได้ถูกบันทึกให้สามารถตั้งค่ารูปแบบโดยอัตโนมัติโดยที่ระบบจะตั้งค่ารูปแบบเมื่อกดบันทึกข้อมูลแล้ว ดังนั้นจึงได้มีการทำ Dynamic Action สำหรับตั้งค่ารูปแบบทันทีเมื่อผู้ใช้งานกรอกข้อมูลเงินเดือน

เงินได้ / เงินหัก (EARNING/DEDUCTION)	ประเภทการจ่าย	วันที่มีผล	วันที่สิ้นสุด	จำนวนเงิน	หมายเหตุ	Action
BAS1 : เงินเดือน	INST	15-09-2018		500000	ตลอดไป	
BAS1 : เงินเดือน	INST	15-08-2018	-	150,000.00		
BAS1 : เงินเดือน	INST	01-08-2018	-	100,000.00		

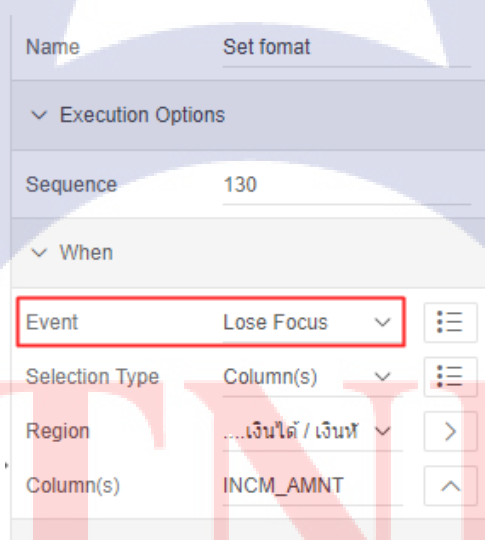
รูปที่ 4.77 รูปแบบของจำนวนเงินก่อนทำ Dynamic Format

- เลือก Column หรือ Item ที่ต้องการ สร้าง Dynamic Action



รูปที่ 4.78 ขั้นตอนการสร้าง Dynamic Action Set format(1)

- Name : ตั้งชื่อให้กับ Dynamic Action
- When Event : Lose Focus



รูปที่ 4.79 ขั้นตอนการสร้าง Dynamic Action Set format(2)

- สร้าง True Action
- Action : Execute PL/SQL

Setting

PLSQL/SQL CODE :

```
:incm_amnt := to_char
(to_number(:INCM_AMNT),
'999,999,999,999.99');
```

Item to Submit : incm_amnt

Item to Return : incm_amnt

รูปที่ 4.80 ขั้นตอนการสร้างDynamic Action Set format(3)

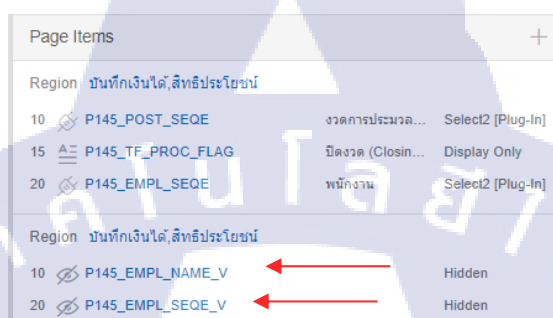
โครงสร้าง (STRUCTURE)	เงินได้ / เงินหัก (EARNING/DEDUCTION)	ลดหย่อนภาษี (TAX ALLOWANCE VAR)	ลดหย่อนภาษี (TAX ALLOWANCE FIX)	ค่าสะสม (YTOD)	ลาออก/หยุดจ่ายเงินเดือน
เงินได้ / เงินหัก (Earning/Deduction)					
เงินได้/เงินหัก	ประเภทการจ่าย	วันที่เริ่ม	วันที่สิ้นสุด	จำนวนเงิน	หมายเหตุ
BAS1 : เงินเดือน	INST	10-09-2018		5,000,000.00	ทดลอง2
BAS1 : เงินเดือน	INST	15-08-2018		150,000.00	
BAS1 : เงินเดือน	INST	01-08-2018		100,000.00	

รูปที่ 4.81 รูปแบบของจำนวนเงินหลังทำ Dynamic Format

4.1.1.1.10 Default ชื่อพนักงาน

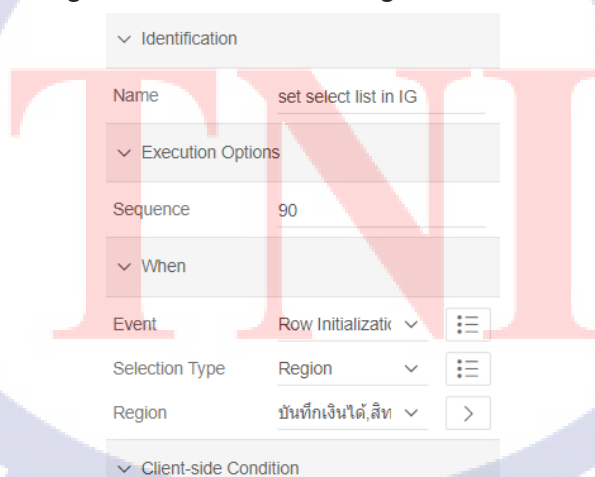
เนื่องจากได้รับ User Requirement ว่า เมื่อทำการเลือกชื่อพนักงานจาก Select list เวลาคลิกปุ่มเพิ่มข้อมูล ให้สามารถแสดงชื่อของพนักงานที่เลือกไว้ทันที แต่ถ้าไม่มีการเลือกพนักงานคนไหน เมื่อคลิกปุ่มเพิ่มข้อมูลจะแสดงค่า Null ไว้

- สร้างไอเทมเพื่อรับค่า PK ของพนักงานและชื่อ



รูปที่ 4.82 ไอเทมรับค่า PK พนักงานและชื่อ

- สร้าง Dynamic Action
- Name : ตั้งชื่อ Dynamic Action
- Event : Row Initialization
- Selection Type : Region
- Region : เลือก Region ที่ต้องการ



รูปที่ 4.83 ขั้นตอนการสร้าง Dynamic Action Set Select list(1)

- สร้าง True Action

Action : Execute JavaScript

Code : var model = this.data.model,
rec = this.data.record,
meta = model.getRecordMetadata(this.data.recordId);
if (meta.inserted) {
model.setValue(rec, "EMPL_SEQE",
{d:\$v("P145_EMPL_NAME_V"), v:
\$v("P145_EMPL_SEQE_V")}); } }



รูปที่ 4.84 ขั้นตอนการสร้าง Dynamic Action Set Select list(2)

- สร้าง Process ขึ้นมา เพื่อส่งค่าเข้า P145_EMPL_NAME_V และ P145_EMPL_SEQE_V



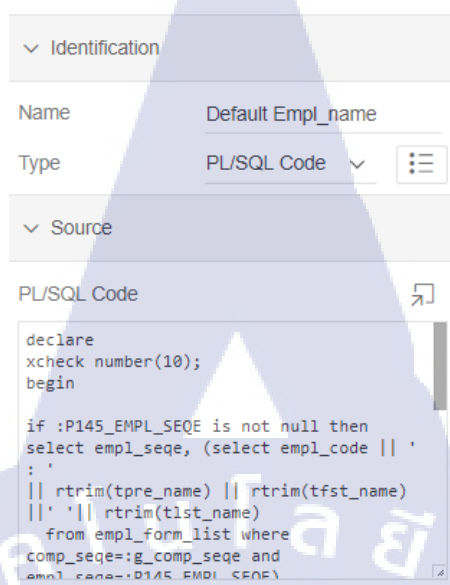
รูปที่ 4.85 ขั้นตอนการสร้าง Process ส่งค่า(1)

- Name : ตั้งชื่อให้ Process

Type :PLSQL Process

PL/SQL CODE :

```
declare
xcheck number(10);
begin
if :P145_EMPL_SEQE is not null then
select empl_seqe, (select empl_code || ' : '
|| rtrim(tpre_name) || rtrim(tfst_name) || ' ' ||
rtrim(tlst_name)
from empl_form_list where comp_seqe=:g_comp_seqe
and empl_seqe=:P145_EMPL_SEQE)
into :P145_EMPL_SEQE_V ,:P145_EMPL_NAME_V
from empl_form_list
where empl_seqe=:P145_EMPL_SEQE;
else
:P145_EMPL_SEQE_V := null;
:P145_EMPL_NAME_V := null;
end if;
end;
```



```

declare
xcheck number(10);
begin

if :P145_EMPL_SEQE is not null then
select empl_seqe, (select empl_code || '
: '
|| rtrim(tpre_name) || rtrim(tfst_name)
|| ' ' || rtrim(tlst_name)
from empl_form_list where
comp_seqe=:g_comp_seqe and
empl_code=:P145_EMPL_SEQE)

```

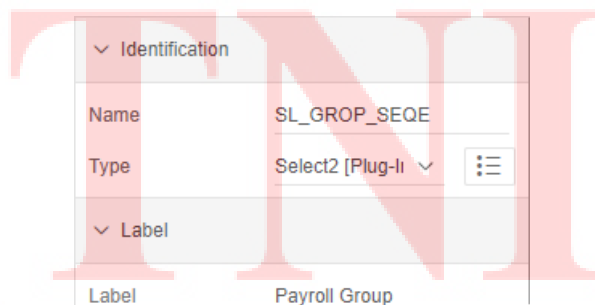
รูปที่ 4.86 ขั้นตอนการสร้างProcessส่งค่า(2)

4.1.1.1.11 Create Button by using PL/SQL for reference the Request Variable

หน้าข้อมูลพนักงานได้มีการสร้าง Select list สำหรับ เลือกกลุ่ม Payroll เพื่ออำนวยความสะดวกแก่ผู้ใช้งาน โดยไม่ต้องย้อนกลับไปหน้าเลือก Payroll Group โดยใช้วิธีส่งค่าตัวแปรRequest ในการเรียกใช้งานProcess ที่กำหนดไว้

- สร้าง Select list บนหน้าข้อมูลพนักงาน

Item เลือกเป็น Select list



รูปที่ 4.87 ขั้นตอนการสร้างSelect list สำหรับเลือกกลุ่มPayroll(1)

- เลื่อนลงมาหัวข้อ List of Values เลือก Type เป็น SQL Query

```
select grop_name d , grop_seqe r from secu_grop_name where
comp_seqe=:g_comp_seqe
order by SYST_GROP
```

▼ List of Values

Type
SQL Query ▼

SQL Query

```
select grop_name d ,
grop_seqe r from
secu_grop_name where
comp_seqe=:g_comp_seqe
order by SYST_GROP
```

รูปที่ 4.88 ขั้นตอนการสร้าง Select list สำหรับเลือกกลุ่ม Payroll(2)

- เลื่อนลงมาหัวข้อ Post text ให้ทำการเขียนโค้ด HTML ในการสร้าง Button

▼ Advanced

CSS Classes

Custom Attributes style="width:200px"

Option HTML Attributes

Pre Text

Post Text

```
<input class="myButton" type="BUTTON"
value="ค้นหา"
onclick="apex.submit('SAVE');">
```

Warn on Unsaved Changes ☒ Page Default ▼

รูปที่ 4.89 ขั้นตอนการสร้างปุ่มส่งค่า Request

- สร้าง Process สำหรับการรับค่า Request ที่ส่งค่ามาจากปุ่มเป็นอันเสร็จสิ้น

▼ Identification

Name Request Process

Type PL/SQL Code ▼

▼ Source

PL/SQL Code

```
declare
begin
If :REQUEST = 'SAVE' then
:G_GROP_SEQE := :SL_GROP_SEQE;
end if;
end;
```

รูปที่ 4.90 ขั้นตอนการสร้าง Process รับค่า Request

Payroll Group: STAFF

Employee ID	Name	Address
AKRST011	วิไลพงษ์ ศรีสุก	SM Tower 21st Floor, 979/52-55 Phaholothin Rd, Samsonmai, Phatut Bangkok 10400
AKRST012	วิไลพงษ์ ศรีสุก	SM Tower 21st Floor, 979/52-55 Phaholothin Rd, Samsonmai, Phatut Bangkok 10400
AKRST013	วิไลพงษ์ ศรีสุก	SM Tower 21st Floor, 979/52-55 Phaholothin Rd, Samsonmai, Phatut Bangkok 10400
AKRST014	วิไลพงษ์ ศรีสุก	SM Tower 21st Floor, 979/52-55 Phaholothin Rd, Samsonmai, Phatut Bangkok 10400
AKRST015	วิไลพงษ์ ศรีสุก	SM Tower 21st Floor, 979/52-55 Phaholothin Rd, Samsonmai, Phatut Bangkok 10400
AKRST016	วิไลพงษ์ ศรีสุก	SM Tower 21st Floor, 979/52-55 Phaholothin Rd, Samsonmai, Phatut Bangkok 10400
AKRST017	วิไลพงษ์ ศรีสุก	SM Tower 21st Floor, 979/52-55 Phaholothin Rd, Samsonmai, Phatut Bangkok 10400
AKRST018	วิไลพงษ์ ศรีสุก	SM Tower 21st Floor, 979/52-55 Phaholothin Rd, Samsonmai, Phatut Bangkok 10400

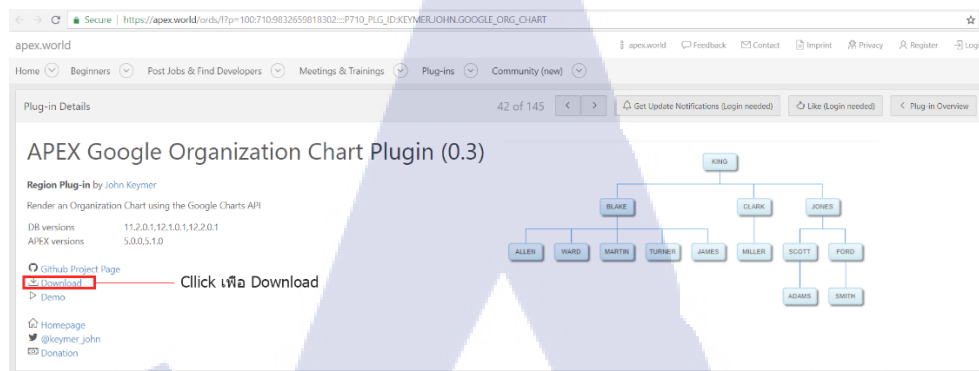
รูปที่ 4.91 หน้าข้อมูลพนักงานที่มีเลือกกลุ่ม Payroll

4.1.2 POC เทคโนโลยีใหม่ๆมาใช้ใน Oracle Apex

4.1.1.2.1 Google Organization Chart

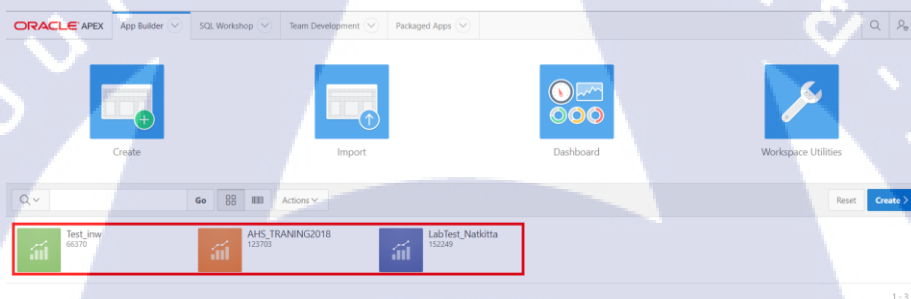
ได้มีการลง Plugin ตัวหนึ่งของ Oracle Apex ช่วยในเรื่องแผนผังระดับชั้นขององค์กร และในอนาคตอาจมีการนำมาใช้ในระบบ Payroll และได้มีการจัดทำเอกสารขึ้นมาด้วย

- Download Plug-in จาก https://apex.world/ords/f?p=APEX_WORLD:HOME



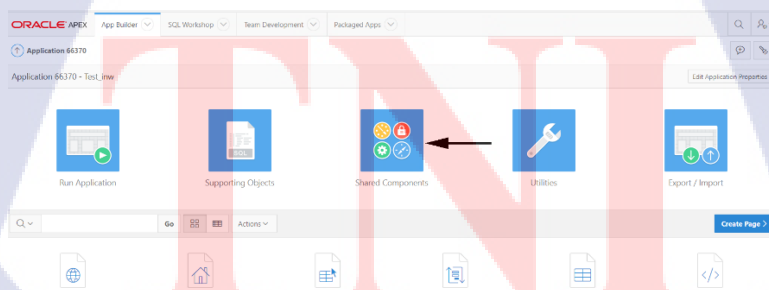
รูปที่ 4.92 ขั้นตอนดาวน์โหลดPlug-in

- เลือก หรือ สร้าง Application ที่ต้องการสร้างสำหรับChart



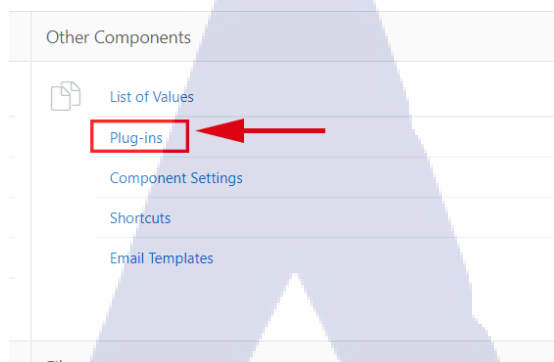
รูปที่ 4.93 ขั้นตอนลงPlug-in Google Organization Chart (1)

- เมื่อเข้ามายังหน้า Application เลือก Shared Components



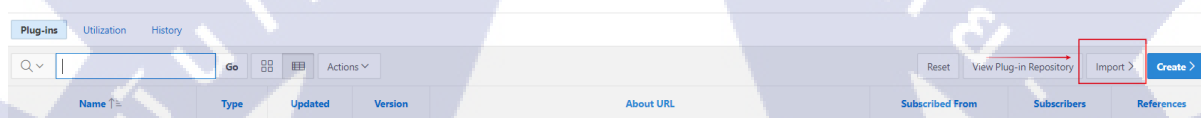
รูปที่ 4.94 ขั้นตอนลงPlug-in Google Organization Chart (2)

- เลือก Plug-in



รูปที่ 4.95 ขั้นตอนลง Plug-in Google Organization Chart (3)

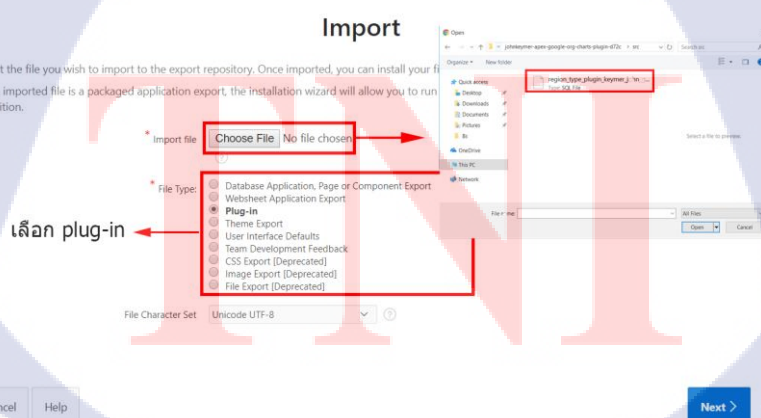
- เมื่อเข้ามาหน้า Plug-ins ให้ทำการเลือก Import



รูปที่ 4.96 ขั้นตอนลง Plug-in Google Organization Chart (4)

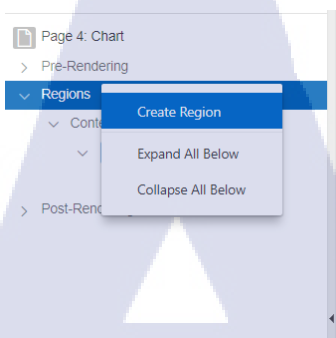
- Import File : ค้นหา Path Download plug-in หาไฟล์เคอร์ SRC
เลือกไฟล์นามสกุล .sql

File Type : เลือก Plug-in



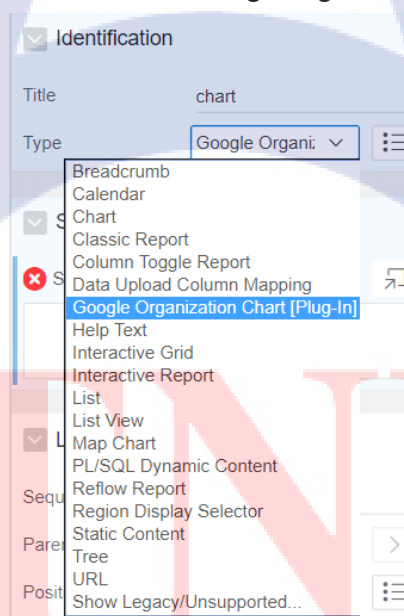
รูปที่ 4.97 ขั้นตอนลง Plug-in Google Organization Chart(5)

- ในหน้า Page Design ของ Chart ให้คลิกขวาตรง Regions เลือก Create Regions



รูปที่ 4.98 ขั้นตอนเรียกใช้ Plug-in Google Organization Chart(1)

- ในส่วน Identification
 Title : กำหนดชื่อ Title
 Type : Google Organization Chart [Plug-In]



รูปที่ 4.99 ขั้นตอนเรียกใช้ Plug-in Google Organization Chart(2)

- ส่วน SQL Query

Select null as link ,to_char(EMP_NO) as key_value , ename as display_value, to_char(MGR) as parent_key from emp

▼ Identification

Title chart

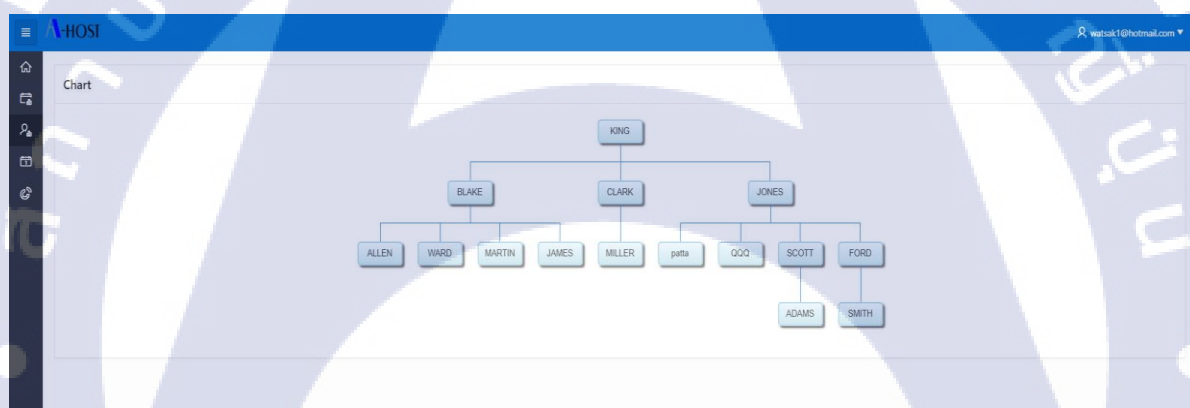
Type Google Organi: ▼

▼ Source

SQL Query

```
select null as link, to_char(empno) as
key_value, ename as display_value,
to_char(mgr) as parent_key from emp
```

รูปที่ 4.100 ขั้นตอนเรียกใช้ Plug-in Google Organization Chart(3)



รูปที่ 4.101 Google Organization Chart

4.1.1.2.2 Notification

มีการทำ POC ในเรื่องของ Notification แสดงข้อความและหายตามเวลาที่กำหนด โดยอนาคตอาจนำมาใช้ในระบบ payroll และได้มีการจัดทำเอกสารขึ้นมาด้วย

- สร้าง Database ขึ้นมา เพื่อเก็บ Notification โดยมีดังนี้
Noti_Name VARCHAR2(40) ชื่อของ NOTIFICATION

Noti_Note NCLOB	คำอธิบายของ NOTIFICATION
Noti_Form TIMESTAMP(6)	
WITH TIME ZONE	เวลาที่ จะ NOTIFICATION ไซว
Noti_Until TIMESTAMP(6)	
WITH TIME ZONE	เวลาที่ จะ NOTIFICATION หาย

- สร้างPage ขึ้นมา เพื่อ Insert Notification เข้า Database โดยตรง Noti_Form Noti_Until ให้เลือกเป็นแบบ DatePicker เพื่อให้สามารถเลือกเวลาได้ และ เทียบเวลาแจบูนัน และ Format

Template
Optional

Template Options
Use Template Defaults

CSS Classes

Format Mask
DD-MON-YY HH24:MI

Width
30 characters

Value Placeholder

รูปที่ 4.102 ขั้นตอนการปรับรูปแบบวันที่

- ส่วนของ Insert Code
begin
insert into
TEST_TASK_SENT_NOTI(NOTI_NAME,NOTI_NOTE,NOTI_FROM,NOTI_TO) values
(NOTI_NAME,;NOTI_NOTE,TO_TIMESTAMP(:DP_FROM,'DD-MON-YY HH24:MI'),TO_TIMESTAMP(:DP_TO,'DD-MON-YY HH24:MI'));
end;

- จากนั้นสร้างPage ใหม่ขึ้นมา เพื่อไว้สำหรับโชว์Notification
สร้าง region ขึ้นมา เป็นแบบ classic Report ใส่ code query ดังนี้

```
select
NOTI_NAME AS ALERT_TITLE ,NOTI_NOTE AS ALERT_DES
from TEST_TASK_SENT_NOTI where (NOTI_FROM is null or
NOTI_FROM <= (SELECT cast(sysdate as TIMESTAMP) "NOW"
FROM DUAL) ) and (NOTI_TO is null or NOTI_TO >= (SELECT
cast(sysdate as TIMESTAMP) "NOW" FROM DUAL) ) order by
NOTI_NAME
```

- เพื่อให้ตัว Notification ของเราสามารถโหลดข้อมูลได้ตลอดเวลาโดยการไม่ต้อง Reload Page ให้นำ Code JavaScript ด้านล่างไปใส่ไว้ใน Header Text ตรง Header and Footer ของ Region Notification

```
<script type="text/javascript">
setInterval("jQuery(‘###REGION_ID#’).trigger(‘apexrefresh’);", 5);
</script>
```

- เพื่อให้ การ Refresh Region ของเราไม่ขึ้น Icon Reload ตลอดเวลา ให้นำ code JavaScript ตามด้านล่างไว้ใน inline ตรง css ของpageที่ต้องการ

```
.u-Processing { display:none }
```

- เพื่อให้Region Notification ของเราปรากฏขึ้นและหายไปตามเวลาที่ตั้งไว้ให้ทำดังนี้

สร้าง Item ขึ้นมาอัน 1 ไว้ที่ Region ที่ไม่ใช่ไว้โชว์ Notification เพื่อเช็คค่า Region Notification ของเรา

จะซ่อนหรือจะโชว์ โดยตั้ง item type ให้เป็นแบบ Hidden

สร้าง Dynamic Action

Event :Click

Selection Type :Region

Region :ชื่อ Region ที่ไม่ใช่ Notification และ เป็น Region ที่อยู่นอกสุดหากมีหลาย Sub Region

True Action

Action : Set value

Setting

Set type :SQL Statement

SQL Code : select NOTI_NAME
from TASK_SEND_NOTI
where (noti_FROM is null or noti_FROM <=
(SELECT cast(sysdate as TIMESTAMP) "NOW"
FROM DUAL))and
(NOTI_TO is null or NOTI_TO >= (SELECT cast(
sysdate as TIMESTAMP) "NOW" FROM DUAL))

Affected Element

Affected Elements : Item(s)

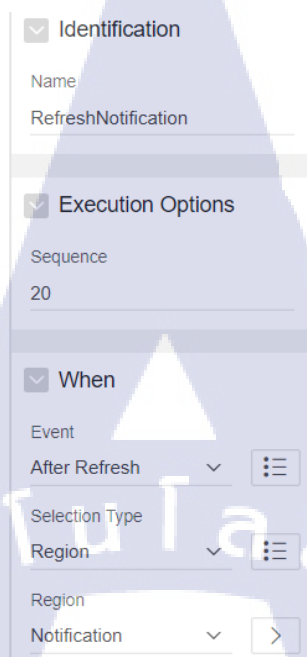
Item(s) : ชื่อ Item ที่เราสร้างขึ้นมา เพื่อเพื่อเช็คค่า Region
Notification ของเราจะซ่อนหรือจะโชว์

- เสร็จแล้วสร้าง Dynamic Action อีกอัน

Event :After Refresh

Selection Type :Region

Region :ชื่อ Region ที่เป็น Notification



☒ Identification

Name
RefreshNotification

☒ Execution Options

Sequence
20

☒ When

Event
After Refresh

Selection Type
Region

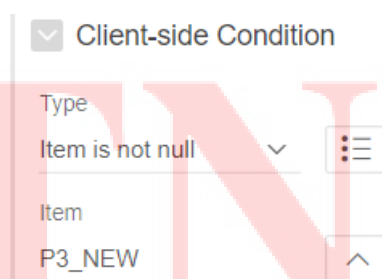
Region
Notification

รูปที่ 4.103 ขั้นตอนการสร้างDynamic Action Refresh(1)

- Client-side Condition

Type : Item is not null

Item : ชื่อ Item ที่เราสร้างขึ้นมาเพื่อเพื่อเช็คค่า Region Notification ของเรา
จะซ่อนหรือจะโชว์ CreateTrue Action เป็น Show



☒ Client-side Condition

Type
Item is not null

Item
P3_NEW

รูปที่ 4.104 ขั้นตอนการสร้างDynamic Action Refresh(2)

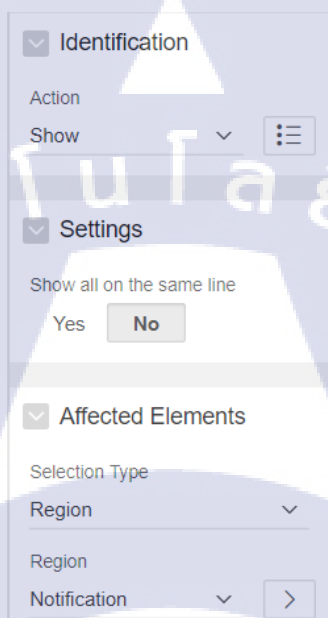
- Create True Action

Action :Show

Affected Element

Selection Type : Region

Region : ชื่อ Region ที่เป็น Notification



รูปที่ 4.105 ขั้นตอนการสร้างDynamic Action Refresh(3)

- Create False Action

Action :Hide

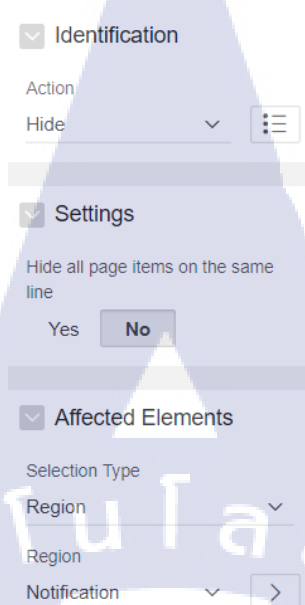
Selection Type :Region

Region :Notification

Affected Element

Selection Type :Region

Region : Region ที่เป็น Notification



▼ Identification

Action

Hide ▼

▼ Settings

Hide all page items on the same line

Yes No

▼ Affected Elements

Selection Type

Region ▼

Region

Notification ▼

รูปที่ 4.106 ขั้นตอนการสร้างDynamic Action Refresh(4)

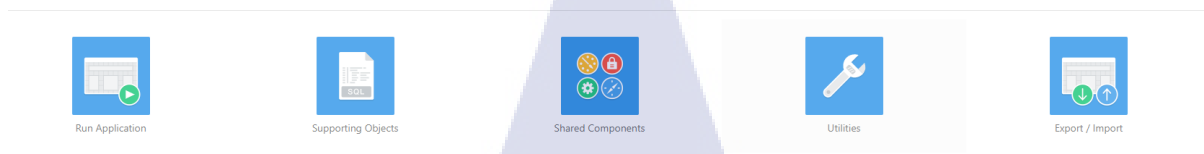
4.1.1.2.3 ToolTip

Poc ในเรื่อง help page และได้พบ Plug in ในส่วนของtooltip และนำ plug In มาใช้งาน สามารถกำหนดข้อความ และกำหนด HTML เข้ามาช่วยได้ และได้มีการจัดทำเอกสารขึ้นมาด้วย

- ดาวน์โหลด Plug-in จาก

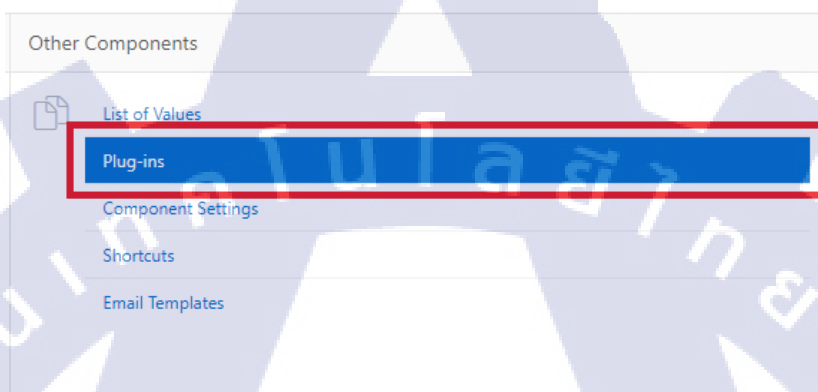
https://apex.world/ords/f?p=100:710:1722780755640::::P710_PLG_ID:DE.DANIELH.APEXTOOLTIP

- เมื่อดาวน์โหลดเสร็จจะได้ไฟล์ Zip ให้ทำการแตกไฟล์ และเข้าไปที่ source > dynamic_action_plugin_de_danielh_apextooltip.sql
- นำไฟล์ Import ลง Apex ไปที่ Shared Component



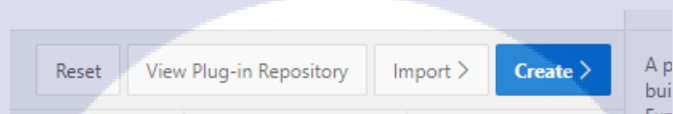
รูปที่ 4.107 ขั้นตอนการลง Plug-in Tooltip (1)

- เลือกหัวข้อ Other Components > Plug-ins



รูปที่ 4.108 ขั้นตอนการลง Plug-in Tooltip (2)

- เลือก Import



รูปที่ 4.109 ขั้นตอนการลง Plug-in Tooltip (3)

- Import File เลือก dynamic_action_plugin_de_danielh_apextooltip.sql
กด Next

Import

Select the file you wish to import to the export repository. Once imported, you can install your file.
If the imported file is a packaged application export, the installation wizard will allow you to run the packaged installation scripts after installing the application definition.

Import file: dynamic_acti...xtooltip.sql

File Type: ☐ Database Application, Page or Component Export ☐ Worksheet Application Export ☒ Plug-in ☐ Theme Export ☐ User Interface Defaults ☐ Team Development Feedback ☐ CSS Export [Deprecated] ☐ Image Export [Deprecated] ☐ File Export [Deprecated]

File Character Set:

รูปที่ 4.110 ขั้นตอนการลง Plug-in Tooltip (4)

- เลือกไอเทมที่ต้องให้แสดง Tooltip

Page 2: Employee Form

> Pre-Rendering

> Regions

> Content Body

> Employee Form

Attributes

> Items

> P2_NF_EMP

> P2_COMM

> P2_DEPT

> P2_TF_EMPNAME

> Validations

> P2_TF_EMPNAME

> P2_HIREDATE

> Validations

> P2_HIREDATE

> Dynamic Actions

> P2_JOB

> P2_NF_SALARY

> Region Buttons

> P2_BUT_SAVE

> PDF

> New

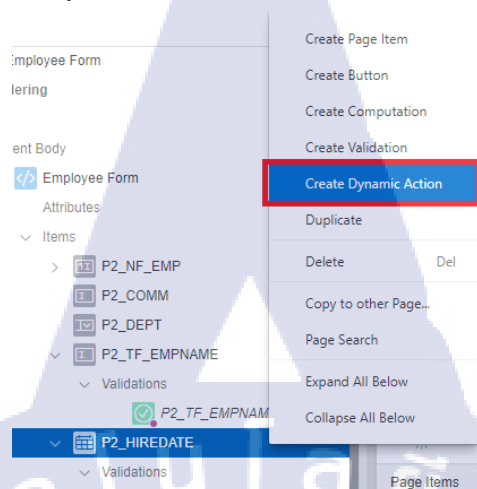
> pdf_region

> Send

> P2_BUT_CLEAR

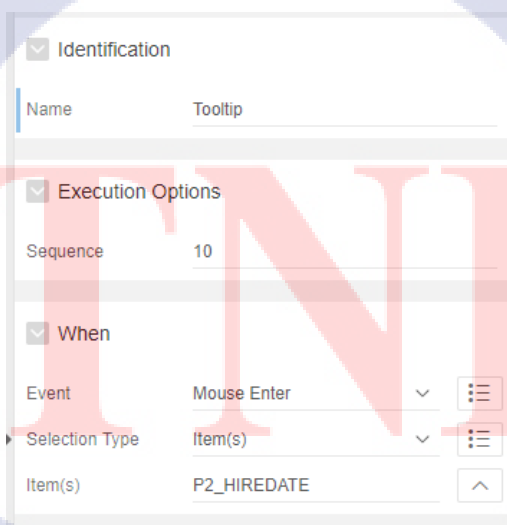
รูปที่ 4.111 ขั้นตอนการเรียกใช้ Tooltip (1)

- สร้าง Dynamic Action



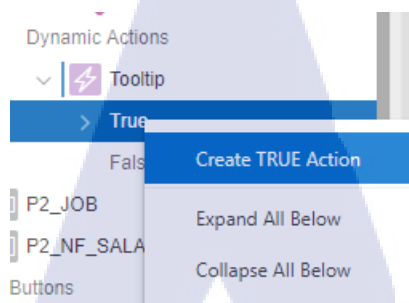
รูปที่ 4.112 ขั้นตอนการเรียกใช้ Tooltip (2)

- Name : ตั้งชื่อให้กับ Dynamic
- Event : Mouse Enter
- Selection Type : Item(s)
- Item(s) : Item ที่ต้องการให้โชว์ Tooltip



รูปที่ 4.113 ขั้นตอนการเรียกใช้ Tooltip (3)

- สร้าง True Action



รูปที่ 4.114 ขั้นตอนการเรียกใช้ Tooltip (4)

- Action : เลือก Apex Tooltip [Plug-in]

Setting

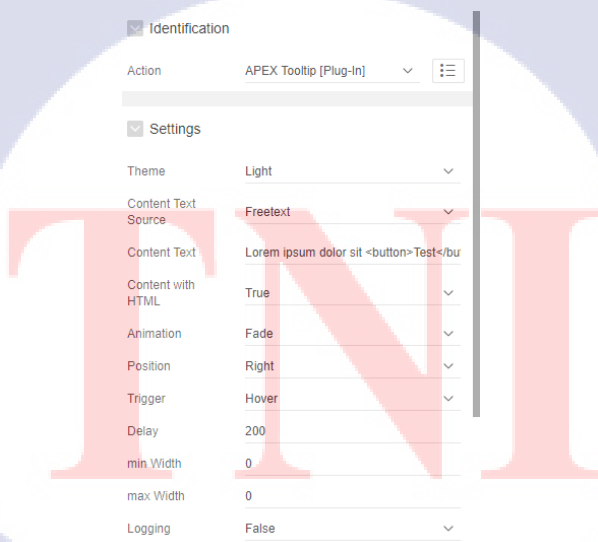
Theme : มีให้เลือกหลากหลายธีม

Context text source :

Freetext กำหนด Text เอง

Text from Item นำข้อความมาจากการกรอกข้อมูล Item ที่กำหนด

Context text : ข้อความที่จะแสดง

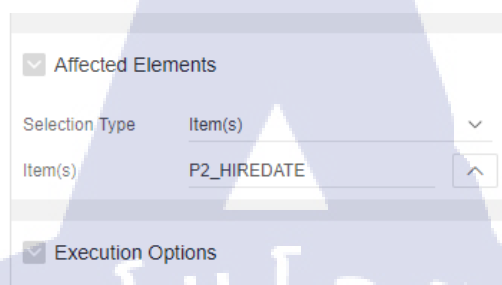


รูปที่ 4.115 ขั้นตอนการเรียกใช้ Tooltip (5)

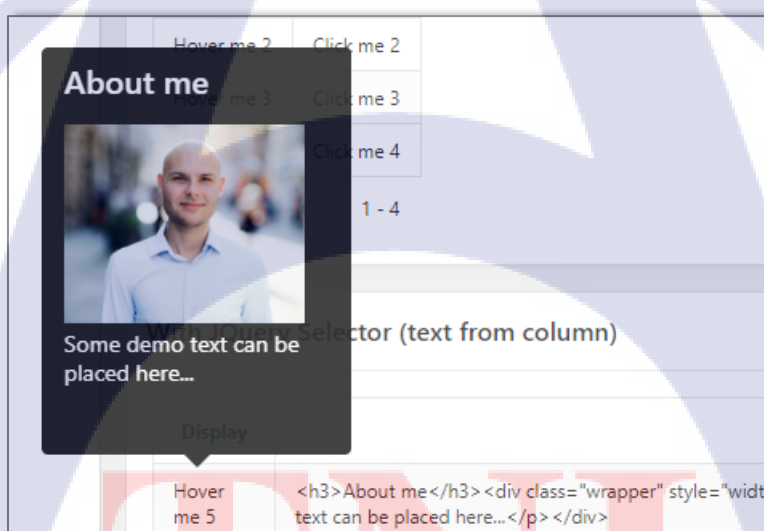
Affected Elements

Selection Type :สามารถเลือกประเภทได้ต่างๆเช่น Item ,
Button , Region

Item(s) :เลือก Item ที่จะทำ



รูปที่ 4.116 ขั้นตอนการเรียกใช้ Tooltip (6)



รูปที่ 4.117 ตัวอย่างTooltip

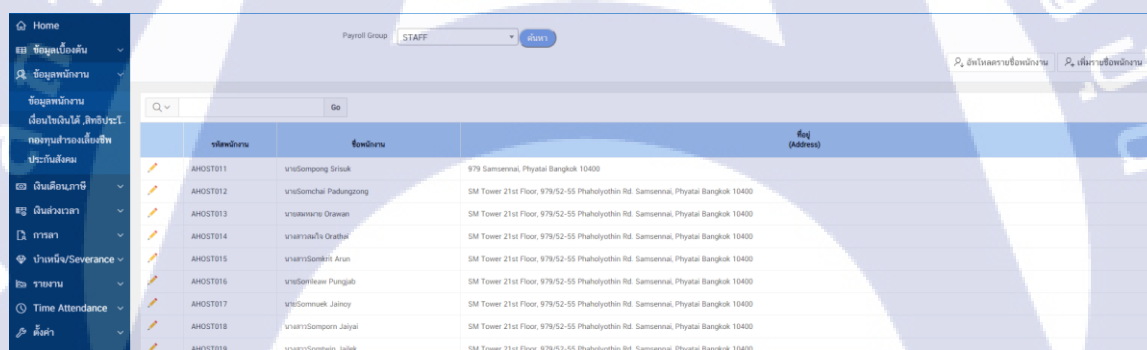
4.1.3 ทดสอบและแก้ไขข้อผิดพลาดของระบบ

ในส่วนของการทดสอบระบบ ได้มีการทดสอบระบบใน DR SITE PRC และการเตรียมข้อมูลให้กับ Enames , สวทท. นำไปทดสอบซึ่งได้มีการทดสอบในส่วนของหน้าพนักงาน การเพิ่มข้อมูลด้วย Import และ Key ผ่านระบบ มีการทดสอบกรณียกยอดสะสมสามารถเช็ค

ข้อมูลค่ายกยอดได้จากการตรวจสอบงวดจากหน้าข้อมูลเงินเดือน และหากพบปัญหาในส่วนใดจะนำมาเป็น issue นำมาแก้ไข เช่น ปัญหาการคำนวณot ใน prc ไม่คำนวณให้อัตโนมัติ ปัญหาพนักงานเข้ากลางเดือนมีเงินเทียบเท่าพนักงานเข้าต้นเดือน เป็นต้น

4.2 ผลการวิเคราะห์ข้อมูล

ตามที่ได้ดำเนินการพัฒนาและสนับสนุนระบบ Payroll โดยใช้ระบบ Oracle Application Express (APEX) โดยมีวัตถุประสงค์ให้ระบบมีความเสถียร แก้ไขปัญหาต่างๆด้านการจัดการบัญชีเงินเดือน คำนวณเงินล่วงเวลา คำนวณภาษี ประกันสังคม กองทุนสำรองเลี้ยงชีพ ฯลฯ เพื่อให้บรรลุวัตถุประสงค์ขององค์กร ให้มีความถูกต้องแม่นยำ ความรวดเร็ว โดยสามารถแบ่งรูปแบบองค์กรเป็นสาขา โครงการ แผนงาน แผนก ฝ่าย และแบ่งตำแหน่งงาน หรือระดับพนักงานได้ สามารถกำหนดเงินได้ เงินหัก ซึ่งในขณะนี้ได้เริ่มทำการทดสอบการใช้งานจริงผ่านการใช้งานของบริษัท Enames โดยผู้ใช้งานแล้ว ระบบสามารถใช้งานได้และแสดงการใช้งาน ได้ดังนี้



รหัสพนักงาน	ชื่อพนักงาน	ที่อยู่ (Address)
AH0ST011	นายSompong Siruk	979 Samsemit, Phayath Bangkok 10400
AH0ST012	นายSomchai Padungzong	SM Tower 21st Floor, 979/52-55 Phaholyothin Rd. Samsemit, Phayath Bangkok 10400
AH0ST013	นายSomchai Oawan	SM Tower 21st Floor, 979/52-55 Phaholyothin Rd. Samsemit, Phayath Bangkok 10400
AH0ST014	นายSomchai Oatthai	SM Tower 21st Floor, 979/52-55 Phaholyothin Rd. Samsemit, Phayath Bangkok 10400
AH0ST015	นายSomchai Anan	SM Tower 21st Floor, 979/52-55 Phaholyothin Rd. Samsemit, Phayath Bangkok 10400
AH0ST016	นายSomchai Pungab	SM Tower 21st Floor, 979/52-55 Phaholyothin Rd. Samsemit, Phayath Bangkok 10400
AH0ST017	นายSomchai Jaiyay	SM Tower 21st Floor, 979/52-55 Phaholyothin Rd. Samsemit, Phayath Bangkok 10400
AH0ST018	นายSompong Jaiyay	SM Tower 21st Floor, 979/52-55 Phaholyothin Rd. Samsemit, Phayath Bangkok 10400
AH0ST019	นายSomchai Jaiyay	SM Tower 21st Floor, 979/52-55 Phaholyothin Rd. Samsemit, Phayath Bangkok 10400

รูปที่ 4.118 ข้อมูลพนักงานของกลุ่มStaff

ภาพนี้แสดงให้เห็นถึงรายชื่อของพนักงานกลุ่มStaff ซึ่งสามารถเลือก กลุ่ม Payroll ที่ต้องการจะเรียกดู และสามารถอัปโหลดรายชื่อพนักงานผ่านไฟล์ CSV รวมถึงสามารถก็ยรายชื่อพนักงานเอง และสามารถกำหนดรายละเอียดวันลาออก วันเริ่มงาน การเปิด ปิด การใช้งานระบบ Apex ผ่านMobile

เงินได้/เงินหัก	ประเภทค่าจ้าง	วันที่เริ่ม	วันที่สิ้นสุด	จำนวนเงิน	หมายเหตุ	Action
BAS1 - เงินเดือน	INST	01-01-2018	-	27,000.00	เงินเดือนปกติ	
BAS1 - เงินโบนัส	INST	09-09-2018	-	25,000.00	Auto-created by pricemppngkpl 14/08/2018	

รูปที่ 4.119 เงินไขเงินได้, สิทธิประโยชน์

ภาพนี้แสดงให้เห็นถึงการกำหนดโครงสร้าง Approve OT Approveการลา เงินได้ เงินหัก ลดหย่อนภาษี วันลาออก และการตรวจสอบค่ายกยอดสะสม ของพนักงานแต่ละคนตามกลุ่มPayrollที่ได้เลือกไว้

ชื่อย่อ	ชื่อ	วันที่คำนวณ	วันที่จ่ายเงิน	โดย บัญชี	โดย พนักงาน	โดย เงินเวลา	คำนวณ ประกันสังคม	คำนวณ กองทุน	คำนวณ ภาษี	บันทึก
sp	sp	01-09-2018	30-09-2018	☒	☒	☒	☒	☒	☒	
August_2018	August_2018	31-08-2018	31-08-2018	☒	☒	☒	☒	☒	☒	
Ytd_7_2018	ยอดรวม_ทุกงวด_ปี_2018_77	31-07-2018	31-07-2018	☒	☐	☐	☐	☐	☐	Y

รูปที่ 4.120 กำหนดงวดการประมวลผล

ภาพนี้แสดงให้เห็นถึงการตั้งงวดการประมวลผลที่นำคำนวณจ่ายเงินเดือน ซึ่งสามารถกำหนดงวดประมวลผลให้คำนวณ 6 แบบ โดยมีดังนี้ โดยบันทึก (Entries) โดยพนักงาน (Recurring) โดยเงินล่วงเวลา (OT) โดยเงินล่วงเวลา (OT) คำนวณภาษี (Taxes Calc) คำนวณประกันสังคม (SSO Calc) คำนวณประกันสังคม (SSO Calc) คำนวณกองทุนสำรองฯ (PF Calc)

วันที่และเวลาจ่ายเงิน	Payroll#	Name	Post name	ตำแหน่งงาน	ขนาดของ	เงินเดือน	สถานะ	แสดงข้อมูล	ลบข้อมูล
11-09-2018 09:19:35	2400	นายสมชาย ใจดี	วิศวกร	1	PAYROLL_RUN	31/08/2018	SUCCESS		
11-09-2018 09:19:35	2360	นายสมชาย ใจดี	วิศวกร	1	PAYROLL_RUN	29/08/2018	SUCCESS		
11-09-2018 09:19:35	2350	นายสมชาย ใจดี	วิศวกร	1	PAYROLL_RUN	28/08/2018	SUCCESS		
11-09-2018 09:19:35	2340	นายสมชาย ใจดี	วิศวกร	1	PAYROLL_RUN	27/08/2018	SUCCESS		
11-09-2018 09:19:35	2330	นายสมชาย ใจดี	วิศวกร	1	PAYROLL_RUN	26/08/2018	SUCCESS		
11-09-2018 09:19:35	2280	นายสมชาย ใจดี	วิศวกร	1	PAYROLL_RUN	25/08/2018	SUCCESS		
11-09-2018 09:19:35	2270	นายสมชาย ใจดี	วิศวกร	1	PAYROLL_RUN	24/08/2018	SUCCESS		

รูปที่ 4.121 ประมวลเงินเดือนและภาษี

ภาพนี้แสดงให้เห็นถึงรายการงวดประมวลที่ได้ประมวลไปแล้ว การแสดงสถานะของงวดประมวลผลว่ามีข้อผิดพลาดหรือไม่ว่าจะโชว์ข้อความ Success และ Fail ซึ่งหน้างวดประมวลผลจะนำงวดการประมวลที่ตั้งงวดขึ้นมาประมวลภาษี เงินเดือน OT ฯลฯ

วันที่และเวลาจ่ายเงิน	Payroll#	Name	ขนาดของ	เงินเดือน	ประเภท	สถานะ	แสดงข้อมูล	ลบข้อมูล	Post page	Post flag
11-09-2018 09:19:51	2180	นายสมชาย ใจดี	PAYROLL_CLOSE_Y	28/08/2018	-	SUCCESS	LOG	Delete	1100	Y
11-09-2018 09:19:51	2180	นายสมชาย ใจดี	PAYROLL_CLOSE_Y	28/08/2018	-	SUCCESS	LOG	Delete	1100	Y
11-09-2018 09:19:51	2080	นายสมชาย ใจดี	PAYROLL_CLOSE_Y	27/08/2018	-	SUCCESS	LOG	Delete	1100	Y
11-09-2018 09:19:51	2050	นายสมชาย ใจดี	PAYROLL_CLOSE_Y	27/08/2018	-	SUCCESS	LOG	Delete	1100	Y
11-09-2018 09:19:51	2040	นายสมชาย ใจดี	PAYROLL_CLOSE_Y	27/08/2018	-	SUCCESS	LOG	Delete	1100	Y
11-09-2018 09:19:51	2020	นายสมชาย ใจดี	PAYROLL_CLOSE_Y	27/08/2018	-	SUCCESS	LOG	Delete	1100	Y
11-09-2018 09:19:51	2010	นายสมชาย ใจดี	PAYROLL_CLOSE_Y	27/08/2018	-	SUCCESS	LOG	Delete	1100	Y
11-09-2018 09:19:51	1990	นายสมชาย ใจดี	PAYROLL_CLOSE_Y	27/08/2018	-	SUCCESS	LOG	Delete	1100	Y

รูปที่ 4.122 ปิดงวดเงินเดือน

ภาพนี้แสดงให้เห็นถึงรายการงวดการประมวลที่ถูกปิดไปแล้วตามแต่ละเดือน โดยงวดที่ถูกปิดไปแล้วจะไม่สามารถแก้ไข อัพเดทข้อมูลได้ และการแสดงสถานะของปิดงวดว่ามีข้อผิดพลาดหรือไม่ว่าจะโชว์ข้อความ Success และ Fail

ลำดับ	รหัส	ชื่อ	เงิน	รายละเอียด
1	AH01ST011	นาย สมพงษ์ ศรีสุข	51,550.00	รายละเอียด
2	AH01ST012	นาย สมชาย พงษ์ทอง	42,275.00	รายละเอียด
3	AH01ST013	นาย สมานันท์ อ้วน	11,875.00	รายละเอียด
4	AH01ST014	นาย สมานันท์ อ้วน	14,250.00	รายละเอียด
5	AH01ST015	นาย สมานันท์ อ้วน	27,895.83	รายละเอียด
6	AH01ST016	นาย สมานันท์ อ้วน	25,031.25	รายละเอียด

รูปที่ 4.123 ข้อมูลเงินเดือน

ภาพนี้แสดงให้เห็นถึงรายละเอียดเงินเดือนของแต่ละพนักงาน โดยสามารถเลือกดูผลการประมวลผลของแต่ละเดือนได้ ว่าเดือนนั้นพนักงานมีข้อมูลเงินเดือนเท่าไร

สรุปได้ว่าโครงการนี้ค่อนข้างประสบผลสำเร็จในการพัฒนาตามแผนงานที่วางไว้ ซึ่งปัจจุบันกำลังอยู่ในช่วงเริ่มต้นใช้งานระบบจริง โดยอาจจะยังติดขัดปัญหาในบางเรื่อง เช่น การศึกษาการใช้งานของ User, ความพึงพอใจของ User แต่ก็มั่นใจว่าสามารถนำมาใช้งานเพื่อตอบสนองได้ในหลายๆ ด้าน ทั้งอำนวยความสะดวกสบาย ความชัดเจนของข้อมูล การทำงานร่วมกับส่วนงานอื่นๆ การจัดเก็บข้อมูล Applications สามารถตอบสนองการทำงานต่างๆ ได้เป็นอย่างดี จึงมั่นใจว่าคุ้มค่ากับการนำระบบนี้มาใช้งานในการดำเนินงาน

4.3 วิเคราะห์และวิจารณ์ข้อมูลโดยเปรียบเทียบผลที่ได้รับกับวัตถุประสงค์และจุดมุ่งหมายในการปฏิบัติงานหรือการจัดทำโครงการงาน

เมื่อทำการวิเคราะห์และเปรียบเทียบผลที่ได้รับกับวัตถุประสงค์ในการปฏิบัติงานแล้ว จะเห็นได้ว่าการนำเอาความรู้ที่ได้รับการอบรมและเรียนรู้มา สามารถนำมาใช้ได้ในการทำงานจริง ซึ่งผลลัพธ์ที่ออกมาอาจจะมีผิดพลาดบ้างเล็กน้อย ขึ้นอยู่กับงานแต่ละชนิด แต่ว่าในบางขั้นตอนก็เกิดเหตุขัดข้อง ทำให้ส่งผลกระทบต่องานจนล่าช้า เช่น การเปลี่ยนแปลง Requirement อยู่สม่ำเสมอ ซึ่งทำให้ยากต่อการนำข้อมูลที่ถูกต้องมาใช้งานจริง ๆ นำมาใช้ เป็นต้น

- 4.2.1 ความสามารถในการใช้ภาษา SQL และภาษา PLSQL ได้เป็นอย่างดี ในการประยุกต์ใช้ในหลายๆจุด เพื่อทำให้งานสำเร็จลุล่วงไปได้ด้วยดี และสามารถนำไปพัฒนาระบบงานในส่วนอื่นๆให้มีความคล่องตัวมากยิ่งขึ้น
- 4.2.2 การศึกษาทำความเข้าใจเกี่ยวกับระบบ Oracle Application Express (APEX) รวมถึงหลักการใช้งานให้เข้าใจมากขึ้น และส่วนที่จะออกแบบหน้าตาระบบ อีกทั้งการดึงข้อมูลมาใช้งานในระบบ
- 4.2.3 การสร้างและแก้ไข Apex ของ Oracle ได้เกือบจะสมบูรณ์ รวมถึงการเข้าใจหลักการทำงานเบื้องต้นที่จำเป็นต้องทราบ
- 4.2.4 การช่วยเหลือและสนับสนุนบริษัท เอ-โฮสต์ จำกัด
- 4.2.5 การนำ Java Script มาประยุกต์ใช้ร่วมกับ Apex เพื่อให้งานสำเร็จลุล่วง และสามารถนำไปพัฒนาระบบงานส่วนอื่นได้
- 4.2.6 ความสามารถในการใช้ CSS แก้ไขปัญหาและการสร้างItemเพิ่ม เพื่อมาสนับสนุนระบบPayroll ให้ตรงตามความต้องการ รวมถึงการตกแต่ง Item

บทที่ 5

บทสรุปและข้อเสนอแนะ

5.1 สรุปผลการดำเนินงาน

จากการที่ได้พัฒนาและทดสอบการใช้งานของลูกค้า ซึ่งได้ผลลัพธ์ดังนี้

- 5.1.1 สามารถเพิ่มข้อมูลพนักงานได้สำเร็จ
- 5.1.2 สามารถกำหนดค่าลดหย่อนและตรวจสอบค่ายกยอดสะสมได้สำเร็จ
- 5.1.3 สามารถบันทึกเงินล่วงเวลา แสดงการเปิด/ปิดงวดได้สำเร็จ
- 5.1.4 สามารถเพิ่มข้อมูลกะพนักงานได้สำเร็จ
- 5.1.5 สามารถแสดงประเภทการลาและเพิ่มข้อมูลประเภทการลาได้สำเร็จ
- 5.1.6 สามารถExport File CSV ในหน้าGenerate Text File ได้สำเร็จ

จากผลลัพธ์ทั้งหมด สามารถสรุปได้ว่า การพัฒนาและสนับสนุนระบบPayroll สามารถตอบสนองความต้องการในด้านOT ข้อมูลพนักงาน ส่งออกไฟล์ข้อมูลเพื่อส่งธนาคารหรือกองทุน เงินได้/เงินหัก Time Attendance ได้เป็นอย่างดี

5.2 แนวทางการแก้ไขปัญหา

ปัญหาที่พบในระหว่างการพัฒนาระบบPayroll ด้วย Oracle Application Express (Apex) ส่วนแรกเป็นการทำความเข้าใจ SQL และ PL/SQL มากขึ้น เนื่องจากตัว Apex เน้นใช้ SQL และ PL /SQL เป็นหลักและอาจมีบางส่วนที่ใช้มากกว่าที่เรียนมา ซึ่งทำให้กระบวนการพัฒนาเป็นไปได้ค่อนข้างช้า แนวทางการแก้ปัญหานี้คือ การทำความเข้าใจจากการศึกษาในหนังสือและอินเทอร์เน็ตประกอบ ก่อนข้างช่วยทำให้เขียน SQL และ PL/SQL ได้ซับซ้อนและขั้นที่สุด

ส่วนปัญหาที่พบอีกอย่างคือ การตั้งค่า Item ร่วมกับ Process ไม่ถูกต้อง หรือไม่เหมาะสม ส่งผลให้การทำงานของระบบ Payroll อาจเกิดปัญหาการใช้งาน ซึ่งแนวทางการแก้ปัญหาคือควรรอบคอบในการตั้งค่า Itemร่วมกับProcessให้ละเอียดก่อน

5.3 ข้อเสนอแนะจากการดำเนินงาน

5.3.1 ควรมีพื้นฐาน SQL และ PL/SQL เนื่องจากตัว Apex การทำงานใช้ SQL PL/SQL เป็นหลัก

5.3.2 ในการทำงานต้องหาข้อมูลและเข้าใจโครงสร้างที่จะลงมือทำเพื่อเตรียมความก่อนลงมือในการทำงานเสมอ

5.3.3 ควรมีพื้นฐานภาษาอังกฤษ เนื่องจาก Apex ข้อมูลภาษาไทยน้อย จำเป็นต้องใช้ภาษาอังกฤษในการสอบถามจากชุมชนในเว็บ Apex

5.3.4 ควรมีพื้นฐาน JavaScript ด้านการกำหนดตัวแปรและกำหนดฟังก์ชัน

TNI

เอกสารอ้างอิง

[1.] Oracle Application Express (Apex) [Online], Available :

<https://apex.oracle.com/en/>

[2.] What Is Application Express? [Online], Available :

<https://www.oracle.com/technetwork/developer-tools/apex/overview/what-is-oracle-apex-3840637.html>

[3.] Oracle SQL Developer Installation [Online], Available :

<https://www.oracle.com/database/technologies/appdev/sql-developer.html>

[4.] Oracle Database Online Documentation 12c Release 1 (12.1) [Online], Available :

https://docs.oracle.com/database/121/nav/portal_5.htm

[5.] พื้นฐานการเขียน JavaScript [Online], Available :

<https://www.tamemo.com/post/120/javascript-fundamental-1-basic/>

[6.] เงินได้ [Online], Available :

<http://www.rd.go.th/m/553.0.html>

[7.] เงินหัก [Online], Available :

<https://www.krungsri.com/bank/th/plearn-plearn/how-withholding-tax-related-with-revenue.html>

[8.] ค่าลดหย่อน [Online], Available :

<http://www.rd.go.th/m/557.0.html>

ประวัติผู้จัดทำโครงการ

ชื่อ – สกุล

นางสาวณัฐกฤตา อภิรมย์เกษม

วัน เดือน ปีเกิด

21 เมษายน 2539

ประวัติการศึกษา

ระดับประถมศึกษา

โรงเรียนแสงอรุณ

ระดับมัธยมศึกษา

โรงเรียนแสงอรุณ

ระดับอุดมศึกษา

คณะเทคโนโลยีสารสนเทศ สาขาเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีไทย-ญี่ปุ่น

ทุนการศึกษา

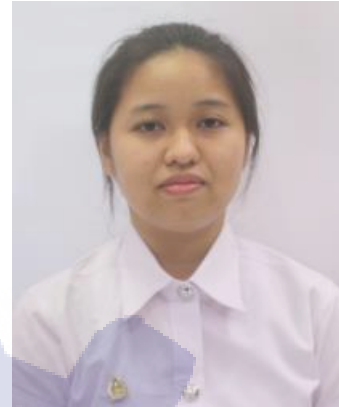
- ไม่มี -

ประวัติการฝึกอบรม

Training Pre-Cooperative Education โครงการสหกิจฯ บริษัท เอ-โฮสต์ จำกัด

ผลงานที่ได้รับการตีพิมพ์

- ไม่มี -



TNI