



การพัฒนา และปรับปรุงระบบเร่งรัด เกี่ยวกับการแจกแจงงาน
กรณีศึกษาของธนาคารเกียรตินาคิน ด้วยเทคโนโลยี พีแอล/เอสคิวแอล

**The Development of Pursuit Application about Assign job for
Kiatnakin Bank Public Company Limited using by PL/SQL**

นายเฉลิมชัย อ่องจรรณู

โครงงานสหกิจนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
ปริญญาวิทยาศาสตรบัณฑิต สาขาเทคโนโลยีมีัลติมีเดีย

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีไทย – ญี่ปุ่น

พ.ศ.2556

การพัฒนา และปรับปรุงระบบเร่งรัด เกี่ยวกับการแจกแจงงาน กรณีศึกษาของธนาคารเกียรตินาคิน
ด้วยเทคโนโลยี พีแอล/เอสคิวแอล

The Development of Pursuit Application about Assign job for Kiatnakin Bank Public Company
Limited using by PL/SQL

นายเฉลิมชัย อ่องจรรยา

โครงการสหกิจศึกษานี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
ปริญญาวิทยาศาสตรบัณฑิต สาขาเทคโนโลยีมีเดีย
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีไทย – ญี่ปุ่น
พ.ศ.2556

คณะกรรมการสอบ

..... ประธานกรรมการสอบ
(อ.ดร.สะพรั่งสิทธิ์ มฤตสาธร)

..... กรรมการสอบ
(อ.อมรพันธ์ ชมกลิ่น)

..... อาจารย์ที่ปรึกษา
(ผศ.นรังสรรค์ วิไลสกุลยง)

..... ประธานสหกิจศึกษาสาขาวิชา
(อ.ภูวดล ศิริทองธรรม)

ลิขสิทธิ์ของสถาบันเทคโนโลยีไทยญี่ปุ่น

ชื่อโครงการ/รายงาน การพัฒนาและปรับปรุงระบบเร่งรัดเกี่ยวกับการแจกแจงงาน
กรณีศึกษาของธนาคารเกียรตินาคิน ด้วยเทคโนโลยี พีแอล/เอสคิว
แอล

The Development of Pursuit Application about Assign job for
Kiatnakin Bank Public Company Limited using by PL/SQL

ผู้เขียน นาย เฉลิมชัย อ่องจรรยา
คณะวิชา เทคโนโลยีสารสนเทศ สาขาวิชาเทคโนโลยีมีัลติมีเดีย
อาจารย์ที่ปรึกษา อาจารย์นันทสรณ์ วิไลสกุลยง
พนักงานที่ปรึกษา นายสุพันธุ์ เจริญสินกุล
ชื่อบริษัท บริษัท เอ -โฮสต์ จำกัด
ประเภทธุรกิจ/สินค้า รับดูแลระบบงาน IT ออกแบบและติดตั้ง

บทสรุป

งานที่ปฏิบัติประกอบไปด้วยการศึกษาระบบ HPAOL/Pursuit การพัฒนาเว็บแอปพลิเคชัน
สนับสนุนการทำงานภายในองค์กร การศึกษาเทคโนโลยีใหม่ๆ และความเป็นไปได้ในการนำเอามา
ปรับใช้เพื่อเพิ่มขีดความสามารถในการทำงาน กระบวนการตรวจสอบ การรับ -ส่งของข้อมูลและ
ออกแบบ ปรับปรุง และพัฒนาเว็บแอปพลิเคชันสำหรับผู้ใช้ เพื่อให้ตรงกับความต้องการของลูกค้า

จากการ ปรับปรุง และพัฒนาเว็บแอปพลิเคชันตามความต้องการของลูกค้า ทำให้มีความรู้
ความเข้าใจในการพัฒนากระบวนการทำงานทางธุรกิจและการพัฒนาเว็บแอปพลิเคชัน และมี
ประสบการณ์การทำงานมากขึ้น ทำให้สามารถทำงานร่วมกับผู้อื่นได้ รู้จักการแก้ไขปัญหาด้วย
ตัวเองก่อนแล้วค่อยถามจากผู้ที่มีประสบการณ์ในกรณีที่เกี่ยวข้องด้วยตัวเองไม่ได้ และมีความพร้อมใน
การทำงานในด้านการพัฒนาซอฟต์แวร์ เว็บแอปพลิเคชัน ภายหลังจากการจบการศึกษาในระดับ
ปริญญาตรี

รูปถ่ายผลงานที่ได้ดำเนินการ

ธนาคาร
เกียรตินาคิน

ธนาคารเกียรตินาคิน จำกัด (มหาชน)

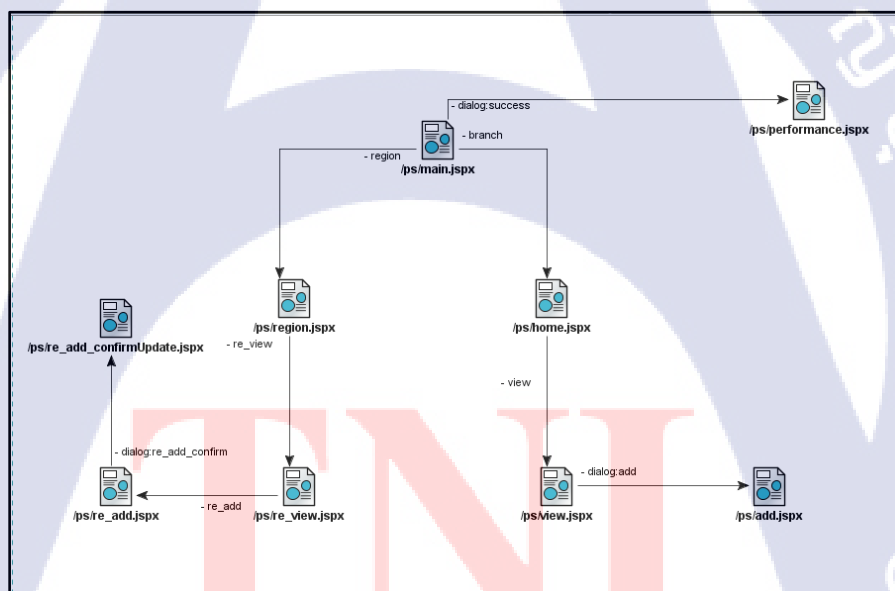
Welcome to Kiattakin Bank

Manage Branch

สาขา: เลือกสาขา สถาน: เลือกสถานะ ภาล: เลือกภาล

แก้ไข % Success	จัดการสาขาในภาล	จัดการสาขาในเขต	ชื่อสาขา	สถานะ	ชื่อสาขาแม่	จังหวัด	% success
Select	จังหวัดสาขา	ชื่อสาขา	สถานะ	ชื่อสาขาแม่	จังหวัด	% success	
☒	0001	สำนักงานใหญ่	สาขาแม่				20
☐	0003	ฉะเชิงเทรา	สาขาแม่	สมุทรปราการ	ฉะเชิงเทรา	20	
☐	0004	ศรีสะเกษ	สาขาแม่	สมุทรปราการ	ฉะเชิงเทรา	50	
☐	0005	เขตหนองคาย	สาขาแม่	นนทบุรี	ฉะเชิงเทรา	10	
☐	0006	นนทบุรี	สาขาแม่	นนทบุรี	ฉะเชิงเทรา	10	
☐	0007	นนทบุรี	สาขาแม่	นนทบุรี	ฉะเชิงเทรา	10	
☐	0008	นนทบุรี	สาขาแม่	นนทบุรี	ฉะเชิงเทรา	20	
☐	0009	นนทบุรี	สาขาแม่	นนทบุรี	ฉะเชิงเทรา	80	
☐	0010	นนทบุรี	สาขาแม่	นนทบุรี	ฉะเชิงเทรา	10	
☐	0011	นนทบุรี	สาขาแม่	นนทบุรี	ฉะเชิงเทรา	10	
☐	0012	นนทบุรี	สาขาแม่	นนทบุรี	ฉะเชิงเทรา	10	
☐	0013	นนทบุรี	สาขาแม่	นนทบุรี	ฉะเชิงเทรา	10	
☐	0014	นนทบุรี	สาขาแม่	นนทบุรี	ฉะเชิงเทรา	10	
☐	0015	นนทบุรี	สาขาแม่	นนทบุรี	ฉะเชิงเทรา	10	
☐	0016	นนทบุรี	สาขาแม่	นนทบุรี	ฉะเชิงเทรา	10	

Copyright © 2014



Project's name	The Development of Pursuit Application about Assign job for Kiatnakin Bank Public Company Limited using by PL/SQL
Writer	Mr. Chaloechai Ongcharun
Faculty	Information of Technology, Technology Multimedia Program
Faculty Advisor	Mr. Narungsun Wilaisakoolyong
Job Supervisor	Mr. Surapun Charoensirinukul
Company's name	A-HOST Co., Ltd.
Business Type / Product	Oracle Core-Technology Products and Advance Services, Hosting, Outsourcing, Enterprise Performance Management and Business Intelligence, System Services

Summary

This Project is to study HPAOL and Pursuit System into web application for support in Organization, new Technology and possibility's Technology to work for design, change and development web application by customer's requirement.

The improvement given by this project results match with customer's requirement and understand business process and have Experience from work and can solve problem with myself before ask supervisor. And finally, we could be able to work in real life.

TNI

กิตติกรรมประกาศ

การที่ข้าพเจ้าได้มาปฏิบัติงานสหกิจศึกษา ณ บริษัทเอ-โฮสจำกัด นับตั้งแต่วันที่ 3 มิถุนายน พ.ศ. 2556 ถึงวันที่ 4 ตุลาคม พ.ศ. 2556 ทำให้ข้าพเจ้าได้รับความรู้และประสบการณ์ต่างๆ ที่มีค่า ต่อการทำงานในอนาคต มากมาย ได้รับความเอาใจใส่ และยังได้รับการอุปการะอย่างดีจากบุคคลหลายๆ ท่านเรื่อยมา สำหรับรายงานวิชาสหกิจศึกษานี้ สำเร็จลงได้ด้วยดีจากความร่วมมือและสนับสนุนจากหลายฝ่าย ดังนั้นข้าพเจ้าขอลงนามขอบคุณไว้ ณ โอกาสนี้

- 1) บิดามารดา บุคคลในครอบครัวและมิตรสหายของข้าพเจ้า
- 2) บุคลากรภายในโครงการสหกิจทุกท่าน
- 3) อาจารย์ในสาขาวิชาเทคโนโลยีสารสนเทศทุกท่าน

และบุคคลท่านอื่นๆ ที่ไม่ได้กล่าวนามทุกท่านที่ได้ให้คำแนะนำช่วยเหลือในการจัดทำรายงาน ข้าพเจ้าใคร่ขอขอบพระคุณผู้ที่มีส่วนเกี่ยวข้องทุกท่าน ที่ให้ข้อมูล เป็นที่ปรึกษาในการทำรายงานฉบับนี้จนเสร็จสมบูรณ์ ตลอดจนให้การดูแลและให้ความเข้าใจเกี่ยวกับชีวิตการทำงานจริง ข้าพเจ้าขอขอบคุณ ไว้ ณ ที่นี้

นายเฉลิมชัย อ่องเจริญ
ผู้จัดทำรายงาน

...../...../.....

สารบัญ

หน้า

บทสรุป.....	ก
Summary	ก
กิตติกรรมประกาศ.....	ง
สารบัญ.....	จ
สารบัญตาราง	ช
สารบัญภาพ	ฉ

บทที่

1. บทนำ.....	1
1.1 ที่ตั้งของสถานประกอบการ	1
1.2 ลักษณะธุรกิจของสถานประกอบการหรือการให้บริการขององค์กร	2
1.3 รูปแบบการจัดองค์กรและการบริการ	6
1.4 ตำแหน่งและหน้าที่งานที่นักศึกษาได้รับมอบหมาย	7
1.5 พนักงานที่ปรึกษา และ ตำแหน่งของพนักงานที่ปรึกษา.....	8
1.6 ระยะเวลาที่ปฏิบัติงาน.....	8
1.7 วัตถุประสงค์หรือจุดมุ่งหมายของการปฏิบัติงานหรือโครงการที่ได้รับมอบหมายให้ ปฏิบัติงานสหกิจศึกษา.....	8
1.8 ผลที่คาดว่าจะได้รับจากการปฏิบัติงานหรือโครงการที่ได้รับมอบหมาย	9

สารบัญ (ต่อ)

หน้า

2. ทฤษฎีและเทคโนโลยีที่ใช้ในการปฏิบัติงาน	10
2.1 Oracle Application Development Framework(Oracle ADF)	10
2.2 Web Application.....	12
2.3 ภาษา SQL (Structured Query Language)	13
2.4 PL/SQL.....	25
2.5 Jdeveloper.....	34
2.6 TOAD For Oracle.....	35
3. แผนงานการปฏิบัติงานและขั้นตอนการดำเนินงาน.....	36
3.1 แผนงานปฏิบัติงาน	36
3.2 รายละเอียดโครงการ.....	36
3.3 ขั้นตอนการดำเนินงานที่นักศึกษาปฏิบัติงานหรือโครงการ	37
4. สรุปผลการดำเนินงาน	107
4.1 ขั้นตอนและผลการดำเนินงาน	107
4.2 ผลการวิเคราะห์ข้อมูล.....	110
4.3 วิจัยข้อมูลโดยเปรียบเทียบผลที่ได้รับกับวัตถุประสงค์และจุดมุ่งหมายการปฏิบัติงาน หรือการจัดทำโครงการ	110

สารบัญ (ต่อ)

หน้า

5. บทสรุปและข้อเสนอแนะ	111
5.1 สรุปผลการดำเนินงาน	111
5.2 แนวทางการแก้ไขปัญหา	111
5.3 ข้อเสนอแนะจากการฝึกงาน	111
เอกสารอ้างอิง	113
ภาคผนวก	114
ก. ภาพถ่ายการปฏิบัติงานสหกิจ ณ บริษัท A-HOST Co.,Ltd.	114
ข. ตัวอย่าง SQL, PL/SQL ที่ปรับปรุงโดยใช้ Toad ในการจัดการในฐานข้อมูล.....	117
ประวัติผู้จัดทำโครงการ	122

สารบัญตาราง

ตารางที่ หน้า

1.1 Task and Assignment.....	7
2.1 แสดง Table : customer1	21
2.2 แสดง Table : customer2	21
2.3 แสดง Table : ผลลัพธ์จากการ UNION	21
2.4 แสดง Table : ผลลัพธ์จากการ UNION ALL	21
3.1 แผนปฏิบัติงานโครงการ	36

TNI

สารบัญภาพประกอบ

ภาพที่ หน้า

1.1 แผนที่ตั้ง A-HOST Company Limited	1
1.2 A-HOST Proud Awards	3
1.3 แผนที่ตั้ง A-HOST Company Limited	6
2.1 แสดงตัวอย่างรูปแบบ JSF	11
2.2 โครงสร้างการเขียนBlock ในPL/SQL แบบ Nested Block	29
2.3 เปรียบเทียบการส่ง PL/SQLและSQL Statement	29
2.4 การสร้าง Function	34
2.5 การสร้าง Procedure	34
2.6 แสดงตัวอย่างโปรแกรม TOAD For Oracle	35
3.1 ขั้นตอนการเริ่มใช้งาน ขั้นตอนที่ 1	37
3.2 ขั้นตอนการเริ่มใช้งาน ขั้นตอนที่ 2	37
3.3 ส่วนประกอบต่างๆ ของโปรแกรม Jdeveloper	38
3.4 ขั้นตอนการตั้งค่า ขั้นตอนที่ 1	39
3.5 ขั้นตอนการตั้งค่า ขั้นตอนที่ 2.....	39
3.6 ขั้นตอนการตั้งค่า ขั้นตอนที่ 3	40
3.7 ขั้นตอนการเชื่อมต่อฐานข้อมูล ขั้นตอนที่ 1	40
3.8 ขั้นตอนการเชื่อมต่อฐานข้อมูล ขั้นตอนที่ 2	41
3.9 ขั้นตอนการเชื่อมต่อฐานข้อมูล ขั้นตอนที่ 3	41
3.10 ขั้นตอนการเชื่อมต่อฐานข้อมูล ขั้นตอนที่ 4	42
3.11 ขั้นตอนการเชื่อมต่อฐานข้อมูล ขั้นตอนที่ 5	43
3.12 ขั้นตอนการเชื่อมต่อฐานข้อมูล ขั้นตอนที่ 6	43

สารบัญภาพประกอบ(ต่อ)

ภาพที่ หน้า

3.13 ขั้นตอนการเชื่อมต่อฐานข้อมูล ขั้นตอนที่ 7	44
3.14 ขั้นตอนการสร้าง Application & Model Project ขั้นตอนที่ 1	44
3.15 ขั้นตอนการสร้าง Application & Model Project ขั้นตอนที่ 2	45
3.16 ขั้นตอนการสร้าง Application & Model Project ขั้นตอนที่ 3	45
3.17 ขั้นตอนการสร้าง Application & Model Project ขั้นตอนที่ 4	46
3.18 ขั้นตอนการสร้าง Application & Model Project ขั้นตอนที่ 5	46
3.19 ขั้นตอนการสร้าง View& Controller ขั้นตอนที่ 1	47
3.20 ขั้นตอนการสร้าง View& Controller ขั้นตอนที่ 2	47
3.21 ขั้นตอนการสร้าง View& Controller ขั้นตอนที่ 3	48
3.22 ขั้นตอนการสร้าง View& Controller ขั้นตอนที่ 4	48
3.23 ขั้นตอนการสร้าง View& Controller ขั้นตอนที่ 5	49
3.24 ขั้นตอนการสร้าง View& Controller ขั้นตอนที่ 6	49
3.25 ขั้นตอนการสร้าง Toplink Map ขั้นตอนที่ 1	50
3.26 ขั้นตอนการสร้าง Toplink Map ขั้นตอนที่ 2	50
3.27 ขั้นตอนการสร้าง Toplink Map ขั้นตอนที่ 3	51
3.28 ขั้นตอนการสร้าง Data Model Object ขั้นตอนที่ 1	51
3.29 ขั้นตอนการสร้าง Data Model Object ขั้นตอนที่ 2	52
3.30 ขั้นตอนการสร้าง Data Model Object ขั้นตอนที่ 3	52
3.31 ขั้นตอนการสร้าง Data Model Object ขั้นตอนที่ 4	53
3.32 ขั้นตอนการสร้าง Data Model Object ขั้นตอนที่ 5	53
3.33 ขั้นตอนการสร้าง Data Model Object ขั้นตอนที่ 6	54

สารบัญภาพประกอบ(ต่อ)

ภาพที่ หน้า

3.34 ขั้นตอนการสร้าง Data Model Object ขั้นตอนที่ 7	54
3.35 ขั้นตอนการสร้าง Data Model Object ขั้นตอนที่ 8	55
3.36 ขั้นตอนการสร้าง Data Model Object ขั้นตอนที่ 9	56
3.37 ขั้นตอนการสร้าง Data Model Object ขั้นตอนที่ 10	56
3.38 ขั้นตอนการสร้าง Data Model Object ขั้นตอนที่ 11	57
3.39 ขั้นตอนการสร้าง Data Model Object ขั้นตอนที่ 12	57
3.40 ขั้นตอนการสร้าง Data Model Object ขั้นตอนที่ 13	58
3.41 ขั้นตอนการสร้าง Configure View & Controller ขั้นตอนที่ 1	58
3.42 ขั้นตอนการสร้าง Configure View & Controller ขั้นตอนที่ 2	59
3.43 ขั้นตอนการสร้าง Configure View & Controller ขั้นตอนที่ 3	59
3.44 ขั้นตอนการสร้างหน้าค้นหาข้อมูล ขั้นตอนที่ 1.....	60
3.45 ขั้นตอนการสร้างหน้าค้นหาข้อมูล ขั้นตอนที่ 2.....	61
3.46 ขั้นตอนการสร้างหน้าค้นหาข้อมูล ขั้นตอนที่ 3.....	61
3.47 ขั้นตอนการสร้างหน้าค้นหาข้อมูล ขั้นตอนที่ 4.....	62
3.48 ขั้นตอนการสร้างหน้าค้นหาข้อมูล ขั้นตอนที่ 5.....	62
3.49 ขั้นตอนการสร้างหน้าค้นหาข้อมูล ขั้นตอนที่ 6.....	66
3.50 ขั้นตอนการสร้างหน้าค้นหาข้อมูล ขั้นตอนที่ 7.....	66
3.51 ขั้นตอนการสร้างหน้าค้นหาข้อมูล ขั้นตอนที่ 8.....	67
3.52 ขั้นตอนการสร้างหน้าค้นหาข้อมูล ขั้นตอนที่ 9.....	68
3.53 ขั้นตอนการสร้างหน้าค้นหาข้อมูล ขั้นตอนที่ 10.....	68
3.54 ขั้นตอนการสร้างหน้าค้นหาข้อมูล ขั้นตอนที่ 11.....	69

สารบัญภาพประกอบ(ต่อ)

ภาพที่ หน้า

3.55 แสดงส่วน input Form ที่ถูกสร้างขึ้น	69
3.56 ขั้นตอนการสร้างหน้าค้นหาข้อมูล ขั้นตอนที่ 12.....	70
3.57 ขั้นตอนการสร้างหน้าค้นหาข้อมูล ขั้นตอนที่ 13.....	70
3.58 ขั้นตอนการสร้างหน้าค้นหาข้อมูล ขั้นตอนที่ 14.....	71
3.59 ผลการ Run ของหน้า search.jspx	71
3.60 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 1	72
3.61 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 2	72
3.62 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 3	74
3.63 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 4	75
3.64 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 5	75
3.65 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 6	76
3.66 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 7	77
3.67 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 8	77
3.68 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 9	78
3.69 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 10	78
3.70 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 11	78
3.71 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 12	79
3.72 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 13	79
3.73 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 14	82
3.74 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 15	83
3.75 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 16	83

สารบัญภาพประกอบ(ต่อ)

ภาพที่ หน้า

3.76 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 17	84
3.77 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 18	87
3.78 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 19	87
3.79 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 20	88
3.80 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 21	88
3.81 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 22	89
3.82 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 23	89
3.83 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 24	90
3.84 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 25	90
3.85 แสดงวิธีการ Run	91
3.86 แสดงการทดลองใช้งานของหน้าค้นหาข้อมูล 1	91
3.87 แสดงการทดลองใช้งานของหน้าค้นหาข้อมูล 2	92
3.88 แสดงการทดลองใช้งานของหน้าค้นหาข้อมูล 3	92
3.89 แสดงการทดลองใช้งานของหน้าค้นหาข้อมูล 4	93
3.90 แสดงการสร้างหน้าลบข้อมูล ขั้นตอนที่ 1	93
3.91 แสดงการสร้างหน้าลบข้อมูล ขั้นตอนที่ 2	94
3.92 แสดงการสร้างหน้าลบข้อมูล ขั้นตอนที่ 3	94
3.93 แสดงการสร้างหน้าลบข้อมูล ขั้นตอนที่ 4	94
3.94 แสดงการสร้างหน้าลบข้อมูล ขั้นตอนที่ 5	95
3.95 แสดงการสร้างหน้าลบข้อมูล ขั้นตอนที่ 6	95
3.96 แสดงการสร้างหน้าลบข้อมูล ขั้นตอนที่ 7	97

สารบัญภาพประกอบ(ต่อ)

ภาพที่ หน้า

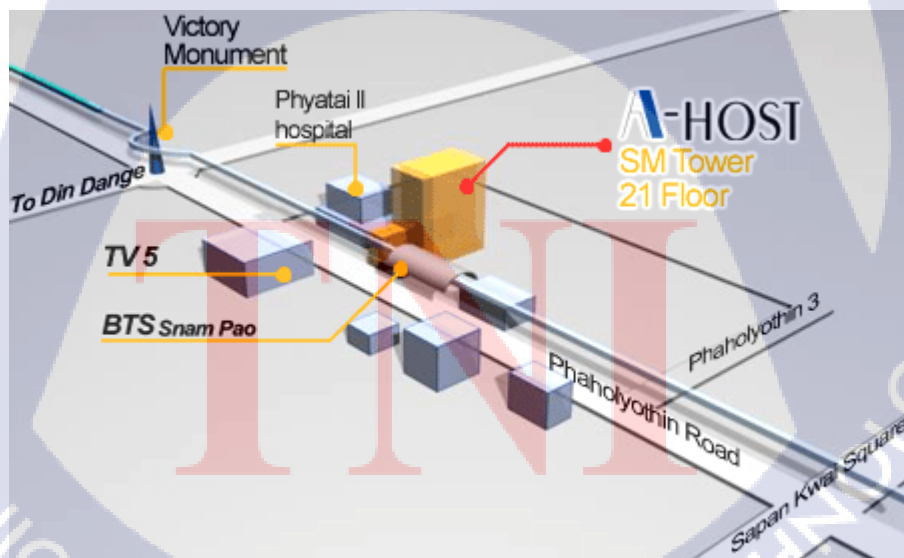
3.97 แสดงการสร้างหน้าลบข้อมูล ขั้นตอนที่ 8	97
3.98 แสดงการสร้างหน้าลบข้อมูล ขั้นตอนที่ 9	98
3.99 แสดงการสร้างหน้าลบข้อมูล ขั้นตอนที่ 10	98
3.100 แสดงการสร้างหน้าลบข้อมูล ขั้นตอนที่ 11	99
3.101 แสดงการสร้างหน้าลบข้อมูล ขั้นตอนที่ 12	99
3.102 แสดงการสร้างหน้าลบข้อมูล ขั้นตอนที่ 13	100
3.103 แสดงการสร้างหน้าลบข้อมูล ขั้นตอนที่ 14	100
3.104 แสดงการสร้างหน้าลบข้อมูล ขั้นตอนที่ 15	101
3.105 แสดงวิธีการรัน	101
3.106 แสดงวิธีการลบข้อมูล ขั้นตอนที่ 1	102
3.107 แสดงวิธีการลบข้อมูล ขั้นตอนที่ 2	102
3.108 แสดงวิธีการลบข้อมูล ขั้นตอนที่ 3	103
3.109 หน้าอินเตอร์เฟซระบบเร่งรัด	104
3.110 หน้าอินเตอร์เฟซจัดการสาขา	104
3.111 ตัวอย่างเอกสาร Change Request 25	105
3.112 ตัวอย่างเอกสาร Test Case	106
ก.1 การปฏิบัติงานขณะอยู่ออฟฟิศ ณ บริษัท A-HOST	115
ก.2 การ Training Oracle ADF 11g	115
ก.3 หมู่ประจําวัน ของการสหกิจศึกษา	116
ก.4 สำนักงานใหญ่ธนาคารเกียรตินาคิน สาขาโอศก	116

บทที่ 1

บทนำ

1.1 ที่ตั้งของสถานประกอบการ

ชื่อสถานประกอบการ	:	A-HOST Company Limited (บริษัท เอ-โฮสต์ จำกัด)
ที่ตั้งของสถานประกอบการ	:	979/53-55 ชั้น 21 ตึก SM Tower ถนนพหลโยธิน แขวงสามเสนใน เขตพญาไท กรุงเทพฯ 10400
โทรศัพท์	:	(66) 2298-0625-32
โทรสาร	:	(66) 2298-0053
E-mail	:	Marketing@a-host.co.th
Website	:	www.a-host.co.th



ภาพที่ 1.1 แผนที่ตั้ง A-HOST Company Limited

1.2 ลักษณะธุรกิจของสถานประกอบการ หรือการให้บริการหลักขององค์กร

บริษัท เอ -โฮสต์จำกัด ได้ก่อตั้งขึ้น ในปี พ.ศ. 2542 เป็นหนึ่งในผู้นำด้านการจัดจำหน่ายผลิตภัณฑ์ซอฟต์แวร์และบริการระบบเทคโนโลยีสารสนเทศในองค์กรแบบครบวงจร โดยเฉพาะการให้บริการแอปพลิเคชันแบบโฮสติ้ง (Hosting) หรือ ASP (Application Service Provider) บริการดูแลระบบ ณ สถานที่ของลูกค้า (IT Outsourcing) รวมถึงบริการให้คำปรึกษาและจัดวางระบบด้วยแอปพลิเคชันสำหรับการวางแผนบริหารจัดการทรัพยากรองค์กร (Enterprise Resource Planning : ERP) ของ Oracle ซึ่งครอบคลุมทั้งองค์กรขนาดใหญ่ องค์กรขนาดกลางและขนาดเล็ก ทั้งนี้ทางบริษัทยังได้รับการแต่งตั้งจาก Oracle ให้ดำเนินกิจการโฮสติ้งในส่วนของบริษัท Oracle Application เป็นรายแรกในประเทศไทยมาตั้งแต่พ.ศ. 2542

ในการดำเนินธุรกิจของเอ-โฮสต์ตลอดระยะเวลา 10 ปี ไม่เพียงแต่ในฐานะผู้บุกเบิกธุรกิจโฮสติ้งและธุรกิจการให้บริการแอปพลิเคชันในรูปแบบ ASP เท่านั้น แต่เอ-โฮสต์ยังได้ทำการติดตั้งระบบไอที รวมทั้งผลิตภัณฑ์ของ Oracle ให้กับลูกค้าจนประสบความสำเร็จมาแล้วเป็นจำนวนมาก ซึ่งหลายรายเป็นหนึ่งในร้อยบริษัทชั้นนำของประเทศไทย แต่ที่สำคัญกว่านั้นก็คือการที่ เอ-โฮสต์ ได้กลายเป็นพันธมิตรทางธุรกิจอย่างแนบแน่นจนกลายเป็นหุ้นส่วนทางกลยุทธ์ และเป็นผู้สนับสนุนสำคัญที่มีส่วนช่วยผลักดันให้ธุรกิจของลูกค้าเติบโตสู่ความสำเร็จ

ปัจจุบันเอ-โฮสต์เป็นหนึ่งในบริษัทลูกของบริษัท ดราก้อนวัน จำกัด ซึ่งเป็นบริษัทมหาชนที่อยู่ในตลาดหลักทรัพย์แห่งประเทศไทย

เป็นเวลากว่า 10 ปีที่เอ-โฮสต์ และ Oracle ได้ดำเนินธุรกิจร่วมกันอย่างใกล้ชิด และถือเป็นพันธมิตรทางธุรกิจกันมา นับตั้งแต่ก่อตั้งบริษัทปีพ.ศ.2542 จวบจนกระทั่งในปัจจุบัน ด้วยความมุ่งมั่นในการนำเสนอผลิตภัณฑ์ของ Oracle ผ่านการให้บริการแอปพลิเคชันในรูปแบบของ ASP ในฐานะที่เอ-โฮสต์เป็นผู้บุกเบิกธุรกิจดังกล่าว และเพิ่มศักยภาพในการดำเนินธุรกิจของลูกค้าได้อย่างเต็มประสิทธิภาพ และเหมาะสม ทำให้ได้รับรางวัลแห่งความสำเร็จและได้รับการยกย่องมาอย่างต่อเนื่อง

ทางเอ-โฮสต์ มีทักษะการทำงานและการบริการที่ครอบคลุมเทคโนโลยีชั้นสูงของ Oracle และได้รับการแต่งตั้งให้เป็น OCAP (Oracle Certified Advantage Partner) รายแรกในไทย ซึ่งเป็นระดับสูงสุดของ Oracle ที่มีให้กับบริษัทคู่ค้าในปี พ.ศ. 2547 และต่อมาในปี พ.ศ. 2549 ได้รับรางวัล “ASEAN partner of the year”



ภาพที่ที่ 1.2 A-HOST Proud Awards

ปัจจุบัน A-HOST มีประเภทของสินค้า และการบริการ ซึ่งสามารถแบ่งออกเป็น 3 กลุ่มใหญ่ๆ ได้แก่

1.2.1 Hosting & Outsource Services

เอ-โฮสต์ ได้ปรับปรุง และขยายการให้บริการ Hosting และ Outsource จนสามารถครอบคลุมความต้องการของลูกค้าได้หลากหลาย โดยเรายึดหลักในการให้บริการที่เรียกว่า “Peace of Mind for the customer” ซึ่งหมายถึงการที่จะทำงานลูกค้าได้อย่างครบวงจร เพื่อให้ลูกค้าจะสามารถใช้งานระบบเทคโนโลยีสารสนเทศที่มีประสิทธิภาพได้อย่างสบายใจ ไร้ความกังวลต่อความเสี่ยงต่างๆ ไม่ว่าจะเป็นเรื่องของปัญหาทางด้านเทคนิค การจัดการระบบและข้อมูลสำรอง การปรับแต่งระบบให้ได้ประสิทธิภาพสูงสุด (Performance Tuning) และที่สำคัญที่สุดคือ การที่เข้ามารับภาระในด้านการบริหารจัดการบุคลากรทางด้านไอทีทั้งหมดแทนลูกค้า

การใช้บริการ Hosting และ Outsource จะทำให้ลูกค้าสามารถทุ่มเทเวลา และทรัพยากรขององค์กรให้กับธุรกิจที่เป็นแกนหลัก (Core Business) โดยทั่วไปแล้วบริการ Hosting และ Outsource ประกอบด้วยส่วนประกอบ และ บริการย่อยๆดังต่อไปนี้

- High Availability and High Performance IT Infrastructure
- Dedicated or Co-Location Service
- Disaster Site
- Oracle E-Business Applications (ERP,CRM,SCM)
- ERP Implementation Service
- System and Database Administration
- Help Desk
- On-Request Services i.e. On-site Support, Software Customization

ทั้งนี้การบริการ Hosting และ Outsource สามารถครอบคลุมได้ทั้งระบบที่ใช้ เทคโนโลยีของOracle และ/หรือ ระบบที่ใช้เทคโนโลยีอื่นๆ

1.2.2 Oracle Core Technology Product and Advanced Services

เอ-โฮสต์เป็นผู้ดำเนินการดำเนินธุรกิจในฐานะ Value-Added Distributor ให้กับ Oracle โดยไม่เพียงแต่ทำหน้าที่ในการจัดจำหน่ายสินค้าในกลุ่ม Core Technology ของ Oracle ทุกประเภท แต่ยังมีทีมผู้เชี่ยวชาญที่จะให้การสนับสนุนและการบริการเสริมอย่างครบวงจรแก่บริษัทลูกค้าและลูกค้า ไม่ว่าจะเป็นการร่วมจัดกิจกรรมทางการตลาด การฝึกอบรม การติดตั้งระบบและการให้คำปรึกษา เพื่อแก้ไขปัญหาต่างๆ

สินค้าบริการที่อยู่ในกลุ่มของ Oracle Core Technology Products and Advanced Services ได้แก่

- Oracle Database and database options
- Oracle Business Intelligence Suite

- Business Partner Development
- System Installation, Intelligence, and Optimization
- Oracle Fusion Middleware (รวมถึง BEA)
- สินค้าอื่นๆ ทุกประเภทของ Oracle
- Marketing and Lead Generation Activities
- SOA-Based Development and Implementation

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

ความต้องการสูงสุดประการหนึ่งของผู้บริหารในการนำเอาระบบไอทีมาใช้องค์กร ไม่ว่าจะเป็นภาครัฐราชการ หรือเอกชน คือ การทำให้ผู้บริหารสามารถได้ข้อมูลที่แสดงให้เห็นถึงสถานะในการดำเนินธุรกิจได้อย่างแม่นยำ รวดเร็ว และนำเอาข้อมูลมาวิเคราะห์และวางแผนทั้งในระดับปฏิบัติการ และในระดับกลยุทธ์ เพื่อให้ธุรกิจสามารถได้เปรียบ ปรับตัวตามสภาพเศรษฐกิจได้ในทุกสถานการณ์ Business intelligence (BI) และ เป็นระบบที่สนองตอบต่อความต้องการในลักษณะดังกล่าวได้เป็นอย่างดีเอ-โฮสต์ มีทีมงานที่ปรึกษาที่มีประสบการณ์ทั้งทางด้านธุรกิจ และทางด้านเทคนิค รวมถึงความเข้าใจในระบบ อย่างลึกซึ้ง จึงทำให้สามารถให้บริการที่ปรึกษาเพื่อออกแบบติดตั้ง เชื่อมโยง และปรับใช้ระบบให้กับลูกค้าอย่างได้ประโยชน์ และประสิทธิภาพสูงสุด อีกทั้งยังมีความยืดหยุ่น และให้การตอบสนองที่เร็ว

1.3 รูปแบบการจัดองค์กรและการบริหารองค์กร



ภาพที่ 1.3 คณะผู้บริหารบริษัท เอ-โฮสต์ จำกัด ของแต่ละแผนก

1.4 ตำแหน่งและหน้าที่งานที่นักศึกษาได้รับมอบหมาย

ตำแหน่งงานที่ได้รับมอบหมายในการปฏิบัติงานสหกิจศึกษาครั้งนี้คือ Programmer โดยจะมีขอบเขตงานและ หน้าที่ที่ได้รับมอบหมายดังตาราง Task and Assignment ดังนี้

ตารางที่ 1.1 Task and Assignment

KPI Description	Expected Results
1. ช่วยสนับสนุนในการทำ HPAOL/Pursuit Change Request	- สามารถช่วยกำหนด Test case ได้ - สามารถทดสอบระบบได้อย่างมีประสิทธิภาพ - สามารถเขียนโปรแกรมได้อย่างมีประสิทธิภาพ (กรณีช่วยเขียน)
2. ทำหน้าที่ผู้ทดสอบระบบ (Tester) ให้กับระบบต่างๆที่เกิดขึ้นของแผนก	- สามารถช่วยกำหนด Test case ได้ - สามารถทดสอบระบบได้อย่างมีประสิทธิภาพ
3. พัฒนา Pursuit Change Request 25 (Re-Assigned) ได้	- โปรแกรมผ่านการทดสอบบนเครื่อง UAT - ทำเอกสารสำหรับการพัฒนาระบบ เช่น Code Diff หรือ programming flow - สามารถช่วยกำหนด test case ได้
4. ทำ ADF Style Template	- สามารถสรุปออกมาได้อย่างน้อย 2 styles - slide presentation - instruction document
5. ศึกษา JavaScript เพื่อสร้าง PDF Form	- สามารถนำไปใช้งานได้จริง - สามารถถ่ายทอดให้ผู้อื่นได้
6. จัดทำ Individual Work List	- นำเสนอ IWL เป็นประจำทุกเดือน

1.5 พนักงานที่ปรึกษา และ ตำแหน่งของพนักงานที่ปรึกษา

พนักงานที่ปรึกษา	:	คุณสุรพันธุ์ เจริญสินกุล
ตำแหน่ง	:	Assistant Consultant
แผนก	:	Academy of Advance Service

1.6 ระยะเวลาที่ปฏิบัติงาน

ระยะเวลาที่ปฏิบัติงานสหกิจศึกษาประมาณ 4 เดือน นับตั้งแต่วันที่ 3 มิ.ย. 2556 – 5 ต.ค. 2556

1.7 วัตถุประสงค์หรือจุดมุ่งหมายของการปฏิบัติงานหรือโครงการที่ได้รับมอบหมายให้ปฏิบัติงานสหกิจศึกษา

การปฏิบัติงานสหกิจในครั้งนี้มีไว้เพื่อเสริมสร้างทักษะ ประสบการณ์และศักยภาพในการทำงานจริงให้แก่ศึกษาก่อนสำเร็จการศึกษาและออกไปสู่ตลาดแรงงานอย่างเป็นระบบ ถือเป็นการพัฒนาทรัพยากรมนุษย์ให้มีมูลค่าเพิ่มขึ้น ภายใต้ความร่วมมือและการดูแลระหว่างมหาวิทยาลัยกับสถานประกอบการหรือองค์กรผู้ใช้นั้นทำให้เกิดการพัฒนาหลักสูตรการเรียนการสอนที่ทันสมัยและมีความสอดคล้องกับความต้องการบุคลากรของตลาดแรงงานสร้างความร่วมมือและความสัมพันธ์ที่ดีระหว่างมหาวิทยาลัยกับ บริษัท เอ เอส จำกัด เพิ่มโอกาสในการได้งานทำและการปรับตัวให้เข้ากับสิ่งแวดล้อมในการทำงาน และเพื่อฝึกฝนให้นักศึกษามีระเบียบวินัยและความรับผิดชอบในหน้าที่ที่ได้รับมอบหมาย

ทั้งนี้กระผมจึงได้มาเป็นส่วนหนึ่งของแผนก Academy of Advance Service ในการพัฒนาระบบของลูกค้าผู้ว่าจ้างคือ ธนาคารเกียรตินาคินซึ่งธนาคารจะมีส่วนกลางในการเพื่อเพิ่มประสิทธิภาพของธุรกิจให้ข้อมูลทำงานอยู่ภายใต้กฎเกณฑ์และระบบบริหารงานขององค์กรที่เป็นผู้ว่าจ้างซึ่งสอดคล้องกับความรู้ที่ได้ทำการศึกษาทั้งจากการศึกษาในมหาวิทยาลัยและจากการฝึกอบรมของทางบริษัทเอ เอส จำกัด

1.8 ผลที่คาดว่าจะได้รับจากการปฏิบัติงานหรือโครงการที่ได้รับมอบหมาย

- นักศึกษาได้รับประสบการณ์ตรงตามสาขา วิชาชีพที่เรียนเพิ่มเติมจากการเรียนในห้องเรียน
- เกิดการเรียนรู้และพัฒนาตนเอง รู้จักทำงานร่วมกับผู้อื่น มีความรับผิดชอบ มีความมั่นใจในตนเองมากขึ้น ซึ่งเป็นคุณสมบัติที่พึงประสงค์ของสถานประกอบการ
- ได้พบกับปัญหาต่างๆ ที่แท้จริงในการทำงาน และคิดค้นวิธีการแก้ปัญหาเฉพาะหน้าได้อย่างถูกต้อง
- เกิดทักษะการสื่อสารข้อมูลการทำงานภายในสถานประกอบการ
- สามารถเลือกสายอาชีพได้ถูกต้องเนื่องจากได้ทราบความถนัดของตนเองมากขึ้น
- สำเร็จการศึกษาเป็นบัณฑิตที่มีศักยภาพในการทำงานและมีโอกาสได้รับการเสนองานก่อนสำเร็จการศึกษา

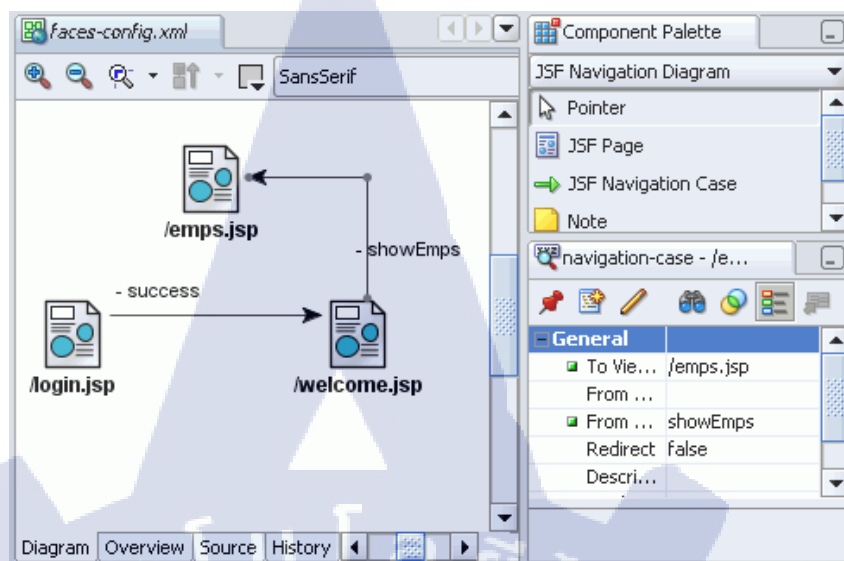
บทที่ 2

ทฤษฎีและเทคโนโลยีที่ใช้ในการปฏิบัติงาน

ในการออกแบบระบบอนุมัติการเช่า-ซื้อรถยนต์ Hire-Purchase Approval Online (HPAOL) และระบบติดตามยึดรถยนต์ (Pursuit) เพื่อให้ผู้ใช้งานระบบของธนาคาร สามารถใช้งาน ได้จริง ลดขั้นตอนในการบันทึกคำเสนอขอเช่าซื้อ ลดเวลาในการตรวจสอบข้อมูลต่างๆ ของลูกค้า และผู้ค้าให้ลดลงโดยการตรวจสอบออนไลน์อัตโนมัติ อีกทั้งช่วยในการบริหารจัดการระบบในการ ขออนุมัติเช่าซื้อและการเร่งรัดหนี้สิน ให้มีความสะดวก รวดเร็ว และง่ายต่อการใช้งาน โดยจะ อธิบายในหัวข้อต่อไปนี้

2.1 Oracle Application Development Framework(Oracle ADF)

ADF เป็น Framework ของทาง Oracle ซึ่งทำงานบน Java EE ช่วยในการพัฒนา Web Application โดยอ้างอิงจากสถาปัตยกรรม MVC ทำให้การพัฒนาซับซ้อนสามารถพัฒนาต่อได้ง่าย นอกจากนี้ยังมีความสามารถพิเศษในการสร้างแบบฟอร์มจาก Input ของฟังก์ชันและตารางจากการ ดูเพียงแค่ผลลัพธ์จากฐานข้อมูล ADF จะทำการผูกค่าเข้ากับกระบวนการทำงานภายในให้อัตโนมัติ ADF ทำงานคู่กับ JSF ซึ่งเป็นการรวมหน้าเว็บเพจหลายๆหน้าเป็นมุมมองเดียวทำให้บริหารจัดการ ได้ง่าย เช่น การใช้ค่ากลางเพียงค่าเดียวสามารถเรียกได้จากทุกหน้าในทันที สามารถกำหนด CSS ตรงกลางเพียงครั้งเดียวสามารถใช้ได้ทุกหน้าภายในวงโดยไม่จำเป็นต้องประกาศบนหน้าเว็บเพจ เป็นต้น นอกจากนี้ยังมีเครื่องมือที่มีเฉพาะใน ADF ในการสร้างตารางให้แสดงจำนวนข้อมูลตาม ขนาดที่ตั้งไว้ หากผลลัพธ์ข้อมูลมีขนาดเกินจะทำการปิดไปเป็นหน้าถัดไปให้



ภาพที่ 2.1 แสดงตัวอย่างรูปแบบ JSF

2.1.1 โครงสร้างของ Oracle ADF

Oracle ADF ใช้สถาปัตยกรรม Model – View – Controller (MVC) โดยรูปแบบการทำงานแบ่งออกเป็นดังนี้

1) Model

ทำหน้าที่รับผิดชอบการจัดการนำข้อมูลขึ้นมาจาก Database และเก็บข้อมูลไว้อีกส่วนหนึ่งคือรับผิดชอบในส่วนของ business logic ที่อยู่ใน Model เช่น การเข้าถึงข้อมูล, การยืนยันความถูกต้องของข้อมูล และ logic การเก็บข้อมูล เป็นต้น หรือบาง framework อาจจะมีในส่วนของ Business Service ด้วยอันนี้ขึ้นอยู่กับมุมมอง

2) View

ทำหน้าที่แสดงผล Graphic User Interface(GUI) เพื่อให้ user สามารถโต้ตอบกับ application ที่เราพัฒนาขึ้นมาได้ และทำหน้าที่รับ input จาก user view ที่พูดถึงถ้าอิงกับการพัฒนา web application ก็อาจเป็นได้ทั้ง HTML,JSP

3) Controller

เป็นตัวสื่อกลางที่ทำงานประสานกันระหว่าง Model และ View
Controller ทำหน้าที่ส่ง request ที่ได้จาก Client แล้วดูว่า request นี้จะส่งไปให้ Model ตัวใด และเมื่อได้ผลลัพธ์จาก Model แล้วก็จะดูต่อว่า response ที่จะได้จะส่งไปให้ View ตัวไหนแสดงผล

2.2 Web Application

web application (เว็บแอปพลิเคชัน) หรือ โปรแกรมประยุกต์บนเว็บ คือโปรแกรมประยุกต์ที่เข้าถึงด้วยโปรแกรมค้นดูเว็บผ่านเครือข่ายคอมพิวเตอร์อย่างอินเทอร์เน็ตหรืออินทราเน็ตเว็บแอปพลิเคชันเป็นที่นิยมเนื่องจากความสามารถในการอัปเดต และดูแล โดยไม่ต้องแจกจ่าย และติดตั้งซอฟต์แวร์บนเครื่องผู้ใช้ ตัวอย่างเว็บแอปพลิเคชันได้แก่ เว็บเมล การพาณิชย์อิเล็กทรอนิกส์ การประมูลออนไลน์ กระดานสนทนา บล็อก วิกี เป็นต้น

การทำงานของ Web Application นั้น โปรแกรมส่วนหนึ่งจะวางตัวอยู่บน Rendering Engine ซึ่งตัว Rendering Engine จะทำหน้าที่หลักๆ คือนำเอาชุดคำสั่งหรือรูปแบบโครงสร้างข้อมูลที่ใช้ในการแสดงผล นำมาแสดงผลบนพื้นที่ส่วนหนึ่งในจอภาพ โปรแกรมส่วนที่วางตัวอยู่บน Rendering Engine จะทำหน้าที่หลักๆ คือการเปลี่ยนแปลงแก้ไขสิ่งที่แสดงผล จัดการตรวจสอบข้อมูลที่รับเข้ามาเบื้องต้น และการประมวลผลบางส่วน แต่ส่วนการทำงานหลักๆ จะวางตัวอยู่บนเซิร์ฟเวอร์

ในลักษณะ Web Application แบบเบื้องต้น ฟังก์ชันเซิร์ฟเวอร์จะประกอบไปด้วยเว็บเซิร์ฟเวอร์ ซึ่งทำหน้าที่เชื่อมต่อกับไคลเอนต์ตามโปรโตคอล HTTP/HTTPS โดยนอกจากเว็บเซิร์ฟเวอร์ จะทำหน้าที่ส่งไฟล์ที่เกี่ยวข้องกับการแสดงผลตามมาตรฐาน HTTP ตามปกติทั่วไปแล้ว เว็บเซิร์ฟเวอร์จะมีส่วนประมวลผลซึ่งอาจจะเป็นตัวแปลภาษา เช่น Script Engine ของภาษา PHP หรือ อาจจะมีการติดตั้ง .NET Framework ซึ่งมีตัวแปลภาษา CLR (Common Language Runtime) ที่ใช้แปลภาษา intermediate จากโค้ดที่เขียนด้วย VB.NET หรือ C#.NET หรืออาจจะเป็น J2EE ที่มีตัวแปลไบต์โค้ดของคลาสที่ได้จากโปรแกรมภาษาจาวา เป็นต้น

2.3 ภาษา SQL (Structured Query Language)

SQL ย่อมาจาก Structured Query Language เป็นภาษาในการสอบถามข้อมูลที่ใช้ในงานด้านการจัดการกับฐานข้อมูลเชิงสัมพันธ์ (Relational Database) โดยเฉพาะ และเป็นภาษาที่มีรูปแบบของคำสั่งคล้ายกับประโยคภาษาอังกฤษจึงเข้าใจได้ง่าย

ภาษา SQL ถูกพัฒนาขึ้นจากแนวคิดของ Relational Calculus และ Relational Algebra เป็นหลัก ภาษา SQL เริ่มพัฒนาครั้งแรกโดย Almaden Research Center ของบริษัท IBM โดยมีชื่อเริ่มแรกว่า “ซีเควล” (Sequel) ต่อมาได้เปลี่ยนชื่อเป็น “เอสคิวเเอล” (SQL) หลังจากนั้นภาษา SQL ก็ได้ถูกนำมาพัฒนาโดยผู้ผลิตซอฟต์แวร์ด้านระบบจัดการฐานข้อมูลเชิงสัมพันธ์จนเป็นที่นิยมกันอย่างแพร่หลายในปัจจุบัน โดยผู้ผลิตแต่ละราย

ก็พยายามที่จะพัฒนาระบบจัดการฐานข้อมูลของตนให้มีลักษณะเด่นเฉพาะขึ้นมา ทำให้รูปแบบการใช้คำสั่ง SQL มีรูปแบบที่แตกต่างกันไปบ้าง เช่น Oracle, Microsoft Access, Sybase, หรือ Microsoft SQL Server เป็นต้น

กลุ่มคำสั่ง SQL สามารถแบ่งได้เป็น 5 ประเภท ดังนี้

1) Retrieval Command เป็นคำสั่งที่ใช้ในการเรียกดูข้อมูล

○ SELECT ใช้ในการเรียกดูข้อมูลที่ต้องการ

2) Data Manipulation Language (DML) เป็นคำสั่งที่ใช้ในการจัดการข้อมูล

○ INSERT ใช้ในการเพิ่มข้อมูล

○ UPDATE ใช้ในการแก้ไขข้อมูล

○ DELETE ใช้ในการลบข้อมูล

3) Data Definition Language (DDL) เป็นคำสั่งที่ใช้ในการสร้าง เปลี่ยนแปลง หรือ ลบ Database object เช่น table, user, view เป็นต้น

○ CREATE ใช้ในการสร้าง Database object

○ ALTER ใช้ในการเปลี่ยนแปลง Database object

○ DROP ใช้ในการลบ Database object

- TRUNCATE ใช้ในการลบข้อมูลทั้ง table ต่างจาก delete ตรงที่ไม่สามารถ rollback กลับมาได้ทำให้ทำงานเร็วกว่า
- 4) Data Control Language (DCL) เป็นคำสั่งที่ใช้ในการกำหนดสิทธิ์หรือถอนสิทธิ์สำหรับการทำงานต่างๆในระบบฐานข้อมูล
 - GRANT ใช้ในการให้สิทธิ์ในการทำงานต่างๆบนฐานข้อมูล
 - REVOKE ใช้ในการถอนสิทธิ์ในการทำงานต่างๆบนฐานข้อมูล
- 5) Transaction Command เป็นคำสั่งที่ใช้ในการจัดการ Database transaction
 - COMMIT ใช้ในการยืนยันการทำงานของ transaction ที่ถูกส่งไปเปรียบเสมือนการ save ข้อมูล
 - ROLLBACK ใช้ในการยกเลิกการทำงานของคำสั่งที่ส่งผ่านไปก่อนหน้านี้เปรียบเสมือนการ undo

2.3.1 การเลือกดูข้อมูล

- **SELECT**

รูปแบบคำสั่ง

SELECT column1, column2, ... FROM table_name

[WHERE conditions];

และ

SELECT * FROM table_name [WHERE conditions];

สำหรับการเลือกดูข้อมูลในทุกคอลัมน์ของ table

- **WHERE Clause**

ใช้ WHERE ต่อท้าย SELECT ในการระบุเงื่อนไข

- **DISTINCT Statement**

ใช้ DISTINCT นำหน้า column นั้น เพื่อเลือกดูข้อมูลโดยไม่ให้แสดงข้อมูลที่มีค่าซ้ำกัน

- **JOIN Clause**

การJOIN เป็นการนำtable ตั้งแต่ 2 tableขึ้นไปมาjoinกัน เพื่อดูข้อมูล
ที่สัมพันธ์เกี่ยวข้องกันซึ่งการJOIN มีอยู่ 2 แบบดังนี้

- **Inner JOIN**

การJOIN แบบนี้จะเลือกเฉพาะข้อมูลมาแสดงเมื่อkey ตรงกัน
เท่านั้น

- **Natural JOIN**

จะใช้กับคอลัมน์ที่มีชื่อและชนิดข้อมูลเหมือนกันใน2table
เพื่อเชื่อมโยงข้อมูลให้อัตโนมัติในกรณีที่มีคอลัมน์ซึ่งมีชนิด
และข้อมูลเหมือนกันมากกว่า 1คอลัมน์ ทุกคอลัมน์เหล่านั้น
จะถูกใช้เป็นตัวเชื่อมโยงร่วมกันทั้งหมด

- **Using JOIN**

จะใช้กับคอลัมน์ที่มีชื่อและชนิดข้อมูลเหมือนกันใน2 table
เพื่อเชื่อมโยงข้อมูลให้อัตโนมัติในกรณีที่มีคอลัมน์ซึ่งมีชนิด
และข้อมูลเหมือนกันมากกว่า 1คอลัมน์ สามารถเลือกได้ว่า
ให้คอลัมน์ใดเป็นตัวเชื่อมโยง

- **JOIN On**

จะใช้กับในกรณีที่ต้องการดูข้อมูลตั้งแต่ 2tableขึ้นไป
และคอลัมน์ที่ใช้สำหรับการเชื่อมโยงไม่จำเป็นต้องมีชื่อ
เดียวกัน แต่ชนิดข้อมูลต้องเหมือนกัน

- **Outer JOIN**

การJOIN แบบนี้จะเลือกข้อมูลมาแสดงโดยที่ไม่ต้องมี key
ตรงกันก็ได้

- **Left Outer JOIN**

ใช้เพื่อระบุว่า ให้แสดงข้อมูลของ table ด้านซ้ายของการ join
ที่ไม่เข้าเงื่อนไขออกมาด้วย กล่าวคือ ให้แสดงข้อมูลของ
table ด้านซ้ายเป็นหลัก

- Right Outer JOIN

ใช้เพื่อระบุว่า ให้แสดงข้อมูลของ table ด้านขวาของการ join ที่ไม่เข้าเงื่อนไขออกมาด้วย กล่าวคือ ให้แสดงข้อมูลของ table ด้านขวาเป็นหลัก

- Full Outer join

ใช้เพื่อระบุว่า ให้แสดงข้อมูลของทั้ง 2 table ที่ไม่เข้าเงื่อนไขของการ join ออกมาด้วย

- **SQL Operators**

- Comparison Operators

=	equal to
<>, !=	is not equal to
<	less than
>	greater than
>=	greater than or equal to
<=	less than or equal to
[not] between...and	
[not] in	
[not] like	
is [not] null	

- Logical Operators

AND, OR and NOT

- **SQL Functions**

- Single row functionsที่มักใช้งาน

- Numeric Functions

CEIL(number)

ใช้ในการปัดเศษขึ้น

FLOOR(number)

ใช้ในการปัดเศษลง

MOD(num1,num2)

ใช้ในการหาค่าmodulation

POWER(num1,num2)

ใช้ในการหาค่ายกกำลัง

ROUND (number,จำนวนทศนิยมที่ต้องการ)

ใช้ในการปัดทศนิยมตามหลักคณิตศาสตร์

TRUNC (number,จำนวนทศนิยมที่ต้องการ)

ใช้ในตัดทศนิยมให้เหลือตามที่ต้องการ นอกนั้นปัดทิ้ง

MAX|GREATEST(value1,value2,...)

หาจำนวนตัวเลขที่มีค่ามากที่สุด

MIN|LEAST(value1,value2,...)

หาจำนวนตัวเลขที่มีค่าน้อยที่สุด

➤ Character Functions Returning Character Values

CONCAT(value1,value2)

ใช้เชื่อมข้อความ

INITCAP(value)

ทำให้อักษรตัวแรกของข้อความเป็นตัวใหญ่

LOWER(value)

ทำให้อักษรทั้งหมดเป็นตัวเล็ก

UPPER(value)

ทำให้อักษรทั้งหมดเป็นตัวใหญ่

TRIM(value)

ใช้ตัดช่องว่างทั้งสองข้าง(ซ้าย,ขวา)

LTRIM(value)

ใช้ตัดช่องว่างด้านซ้าย

RTRIM(value)

ใช้ตัดช่องว่างด้านขวา

REPLACE(value1,value2)

ใช้ในนำคำหนึ่งมาแทนที่อีกคำหนึ่ง

SUBSTR(value,ตำแหน่งที่เริ่ม,จำนวนตัวอักษรทั้งหมด)
ใช้ในการตัดคำออกจากข้อความ

➤ Character Functions Returning Number Values

INSTR(value,อักขระที่ต้องการหา)
ใช้ในการหาตำแหน่งของตัวอักษรที่ต้องการจากคำนั้นๆ
LENGTH(value)
ใช้ในการนับจำนวนตัวอักษรทั้งหมดของข้อความนั้นๆ

➤ Datetime Functions

ADD_MONTHS(date,จำนวนเดือนที่จะเพิ่ม)
ใช้เพิ่มเดือนให้กับวันที่ที่กำหนด
EXTRACT (date)
ใช้ตัดวัน หรือ เดือน หรือปี ออกมาจากวันเวลาที่กำหนด
LAST_DAY(date)
หาวันสุดท้ายของเดือนนั้นๆ
MONTHS_BETWEEN(date1,date2)
หาจำนวนเดือนระหว่างวันที่ที่ระบุ
NEXT_DAY(date,วันที่ต้องการหาวันที่)
ใช้หาวันที่จากวันที่ระบุ เช่น หากวันนี้เป็น 1 ม.ค. 2555 แล้ว
วันศุกร์จะเป็นวันที่เท่าไร

➤ Conversion Functions

CAST(column as datatype)
ใช้แปลงชนิดข้อมูล
TO_CHAR (character)
ใช้แปลงตัวอักษรให้อยู่ในรูปแบบที่ต้องการ
TO_CHAR (datetime)
ใช้แปลงวันที่ให้เป็นตัวอักษร และจัดรูปแบบวันที่ได้
TO_CHAR (number)
ใช้แปลงตัวเลขให้เป็นตัวอักษร และจัดรูปแบบตัวเลขได้
TO_DATE(character)

ใช้แปลงตัวอักษรให้เป็นวันที่

TO_NUMBER(character)

ใช้แปลงตัวอักษรให้เป็นตัวเลข

➤ Miscellaneous Single-Row Functions

CASE..WHEN

เป็นคำสั่งที่ใช้สำหรับการกำหนดเงื่อนไขการแสดงค่าของ
คอลัมน์ หรือค่าของ Value เหมือนกับการใช้ IF...Else...

รูปแบบคำสั่ง

CASE input_expression

WHEN when_expression1 THEN result_expression1

WHEN when_expression2 THEN result_expression2

...

ELSE result_expression

END;

DECODE

เป็นคำสั่งที่ใช้สำหรับการกำหนดเงื่อนไขเช่นเดียวกับ

case..when

รูปแบบคำสั่ง

DECODE(input_expression, when_expression1,

result_expression1, when_expression2,

result_expression2,...,result_expression)

NVL(column,ค่าที่จะนำมาแทนnull)

ถ้าค่าใดเป็นnullในคอลัมน์นั้น จะถูกแทนที่ด้วยค่าที่กำหนด

O Group functions/Aggregate functions ที่มักใช้งาน

SUM(column_name)

ใช้ในการหาผลรวมของทุกแถวในคอลัมน์นั้นๆ

AVG(column_name)

ใช้ในการหาค่าเฉลี่ยของทุกแถวในคอลัมน์นั้นๆ

MAX(column_name)

ใช้ในการหาที่มากที่สุดจากทุกแถวในคอลัมน์นั้นๆ

MIN(column_name)

ใช้ในการหาค่าที่น้อยที่สุดจากทุกแถวในคอลัมน์นั้นๆ

COUNT(column_name)

ใช้ในการนับจำนวนแถวในคอลัมน์นั้นๆ

- **Subquery**

SUBQUERY คือการใช้คำสั่ง SELECT ซ้อนคำสั่ง SELECT

การค้นหาข้อมูลโดยการใช้ SUBQUERY นั้นจะทำให้เราสามารถค้นหาข้อมูลในรูปแบบที่ซับซ้อนมากขึ้นได้เพราะว่าเป็นการค้นหาข้อมูลที่เราได้ค้นหามาแล้วนำมาใช้เป็นเงื่อนไขให้กับอีกผลการค้นหาหนึ่ง

โดยคำสั่ง SELECT ที่เป็นเงื่อนไขจะอยู่ภายในวงเล็บ

ตัวอย่างการใช้งาน

SELECT customer_number

FROM purchases

WHERE price > (SELECT avg(price) + 100 FROM purchases);

- **UNION และ UNION ALL**

ทั้ง UNION และ UNION ALL เป็นคำสั่งที่ใช้สำหรับการเลือกข้อมูล

โดยทำการรวมจำนวนแถวระหว่าง Table เข้าด้วยกัน สำหรับ UNION

และ UNION ALL ต่างกันตรงที่ UNION จะเลือกข้อมูลระหว่าง 2 ตาราง

ตามกฎ DISTINCT ของ Table นั้น ๆ คือข้อมูลใน Table หนึ่ง ๆ จะไม่ซ้ำ

กัน ส่วน UNION ALL เลือกข้อมูลโดยไม่สนใจว่าจะซ้ำกันหรือไม่

คือเลือกเอาทั้งหมดนั่นเอง

ตัวอย่าง

สมมุติมีข้อมูล 2 table ดังนี้

ตารางที่ 2.1 แสดง Table : customer1

CustomerID	Name	CountryCode	Budget
C001	Win Weerachai	TH	1000000
C002	John Smith	EN	2000000

ตารางที่ 2.2 แสดง Table : customer2

CustomerID	Name	CountryCode	Budget
C001	Win Weerachai	TH	1000000
C002	John Smith	EN	2000000
C002	John Smith	EN	2000000

ตารางที่ 2.3 แสดง Table : ผลลัพธ์จากการ UNION

CustomerID	Name	CountryCode	Budget
C001	Win Weerachai	TH	1000000
C002	John Smith	EN	2000000

ตารางที่ 2.4 แสดง Table : ผลลัพธ์จากการ UNION ALL

CustomerID	Name	CountryCode	Budget
C001	Win Weerachai	TH	1000000
C002	John Smith	EN	2000000
C002	John Smith	EN	2000000

2.3.2 การใช้คำสั่ง insert, update, delete ในการจัดการกับข้อมูล

- **INSERT**

รูปแบบคำสั่ง

```
INSERT INTO table_name (column1, column2, column3,...)
VALUES (value1, value2, value3,...)
```

และ

```
INSERT INTO table_name VALUES (value1, value2, value3,...)
```

สำหรับเพิ่มค่าให้กับทุกคอลัมน์ของ table

- **UPDATE**

รูปแบบคำสั่ง

```
UPDATE table_name
SET column1=value, column2=value2,...
WHERE some_column=some_value;
```

* หากไม่ได้ WHERE conditions ให้กับ UPDATE จะเป็นการแก้ไขข้อมูล
ในทุก row ของ table

- **DELETE**

รูปแบบคำสั่ง

```
DELETE FROM table_name
WHERE some_column=some_value;
```

* หากไม่ได้ WHERE conditions ให้กับ DELETE จะเป็นการลบข้อมูล
ในทุก row ของ table

2.3.3 การ create, alter, และ drop table

- **CREATE Table**

รูปแบบคำสั่ง

```
CREATE TABLE table_name
(
column_name1 data_type [Constraints],
column_name2 data_type [Constraints],
column_name3 data_type [Constraints],
....
)
```

constraints มีดังนี้

- ☐ NOT NULL
- ☐ UNIQUE
- ☐ PRIMARY KEY
- ☐ FOREIGN KEY
- ☐ CHECK
- ☐ DEFAULT

- **CREATE Table with Subquery**

รูปแบบคำสั่ง

```
CREATE TABLE table_name
AS
SELECT column1, column2, ... FROM table_name;
```

- **ALTERTable**

- **ADD**

รูปแบบคำสั่ง

```
ALTER TABLE table_name
add (column_namedatatype(size));
```

- **MODIFY**

รูปแบบคำสั่ง

```
ALTER TABLE table_name
MODIFY (column_namedatatype(size));
```

- **RENAME**

รูปแบบคำสั่ง

```
ALTER TABLE table_name
RENAME COLUMN old_name to new_name;
```

- **DROPTable**

รูปแบบคำสั่ง

```
DROP TABLE table_name;
```

2.3.4 การ create และ drop view

- **CREATE**

รูปแบบคำสั่ง

```
CREATE [OR REPLACE] VIEW view_name
AS
SELECT column1, column2, ... FROM table_name;
```

- **DROP**

รูปแบบคำสั่ง

```
DROP VIEW view_name;
```

2.3.5 การใช้คำสั่ง commit และ rollback

- COMMIT

โดยปกติแล้ว ระบบฐานข้อมูล RDBMS จะไม่ยืนยันการทำงาน transaction อย่างอัตโนมัติ หากใช้คำสั่ง INSERT, UPDATE, DELETE แล้วปิดการใช้งานไป โดยไม่ได้ COMMIT ก็จะทำให้ transaction เหล่านั้นหายไป ไม่ได้มีการบันทึกใดๆเกิดขึ้นทั้งสิ้น เพราะฉะนั้นทุกๆครั้งหลังใช้คำสั่งประเภท DML จำเป็นต้อง COMMIT ทุกครั้ง แต่หากใช้งานคำสั่งประเภท DDL เช่น CREATE เป็นต้น ไม่จำเป็นต้อง COMMIT เพราะมันจะ COMMIT ให้โดยอัตโนมัติ ดังนั้นหากใช้คำสั่งประเภท DML ไปแล้วปิดท้ายด้วยคำสั่งประเภท DDL ก็ไม่ต้อง COMMIT

ROLLBACK

ROLLBACK เป็นคำสั่งที่ใช้ในยกเลิกการทำงานของคำสั่งที่ส่งผ่านไปก่อนหน้านี้ เช่น การส่งคำสั่งเพื่อลบข้อมูลบน table “employee”

(DELETE * FROM employee;) แล้วต้องการยกเลิกการลบข้อมูล ทำได้โดยใช้คำสั่ง

ROLLBACK ทันทีภายหลังคำสั่ง DELETE ผลจากการใช้คำสั่ง ROLLBACK จะทำให้ได้ข้อมูลที่ลบด้วยคำสั่ง DELETE กลับคืนมา

แต่อย่างไรก็ตาม ROLLBACK มีข้อจำกัด คือ ไม่สามารถเรียก transaction ที่เกิดจากคำสั่ง

DROP (TABLE, INDEX), ALTER (TABLE, USER), CREATE (TABLE, VIEW)

2.4 PL/SQL

PL/SQL เป็นภาษาโปรแกรมสำหรับเครื่องมือพัฒนาซอฟต์แวร์ของ Oracle ที่ถูกออกแบบมาสำหรับใช้เพื่อเข้าถึงและจัดการข้อมูลในฐานข้อมูล RDBMS PL ย่อมาจาก Procedure Language เพื่อเป็นส่วนเพิ่ม (Extension) ใน Standard SQL ให้ทำงานแบบ Procedural กับฐานข้อมูล PL/SQL จึงเป็นตัวเชื่อมระหว่าง Database Technology และ Procedural programming language, PL/SQL เป็นเครื่องมือใช้พัฒนาระบบงานที่เพิ่มความสามารถให้กับ SQL ของ Oracle ทำให้พัฒนาระบบงานที่สลับซับซ้อน และเข้าถึงจัดการข้อมูลในฐานข้อมูลได้เป็นอย่างดี

ดีด้วยความสามารถของ PL/SQL เราสามารถใช้คำสั่ง DML(Data Manipulate Language) เช่น Insert, Delete, Update, Select กับข้อมูล ใช้คำสั่ง Loop ทำงานแบบวนซ้ำ ใช้ If และ Case ตรวจสอบเงื่อนไข

โปรแกรมที่เขียนขึ้นด้วย PL/SQL นั้นจะประกอบขึ้นด้วยสิ่งที่เราเรียกว่า Block โดยภายใน Block หนึ่งๆสามารถมี Blockย่อยอยู่ข้างในได้ หรือเรียกว่า Nested Block

โครงสร้างของการเขียน PL/SQL

DECLARE –จะมีหรือไม่มีก็ได้

พื้นที่สำหรับประกาศ ตัวแปรต่าง ๆ และ cursors

BEGIN -- จำเป็นต้องมีเสมอ

พื้นที่ตรงนี้สำหรับ code ที่ใช้ในการประมวลผล

EXCEPTION -- จะมีหรือไม่มีก็ได้

END; -- จำเป็นต้องมีเสมอ

หลักการตั้งชื่อตัวแปรของ PL/SQL

- 1) ต้องขึ้นต้นด้วยตัวอักษรเท่านั้น
- 2) สามารถใช้ตัวอักษรหรือตัวเลขในการตั้งชื่อตัวแปร
- 3) สามารถใช้อักขระพิเศษได้เช่น \$, _ เป็นต้น
- 4) ความยาวของตัวแปรต้องไม่เกิน 30 ตัวอักษร
- 5) ชื่อตัวแปรต้องไม่ไปซ้ำกับคำสงวนของ PL/SQL เช่น begin , end

ประเภทของการเขียน PL/SQL

- 1) Anonymous Block

เป็น Block ที่ไม่มีชื่อ ไม่เก็บข้อมูลที่ได้จากการประมวลผลลง Database สามารถเรียกใช้โปรแกรมอื่นได้ แต่ไม่สามารถเรียกใช้งานตัวเองได้

2) Named Block

จะเป็น Subprogram ต่างๆ เช่น Function, Procedure

มีการประมวลผลบน Database และจัดเก็บข้อมูลลง Database ด้วย

การใส่ Comment ใน PL/SQL

1) Single Line ใช้ --comment

2) Multiple Line ใช้ /*comment*/

การประกาศใช้งานตัวแปรและกำหนดค่าเริ่มต้น

SYNTAX :

```
variable_name [CONSTANT] datatype [NOT NULL] [{DEFAULT|:=} initial_value];
```

โดย :

variable_name	คือชื่อตัวแปร
CONSTANT	เป็น keyword ว่าตัวแปรนี้ เป็น constant variable (ค่าคงที่) ไม่สามารถเปลี่ยนค่าได้
datatype	คือประเภทของตัวแปร เช่น NUMBER[(p[,s])] number (p = precision, s = scale) CHAR[(n)] fixed length character (default = 1 char) VARCHAR2(n) variable length character n คือ maximum length BOOLEAN logical มี 3 ค่า คือ (True, False, Null)
NOT NULL	เป็นการตั้งกฎไว้ว่า ตัวแปรนี้ห้ามเป็น NULL
{DEFAULT :=} initial_value	เป็นการกำหนดค่าเริ่มต้นให้ตัวแปร

หมายเหตุ ถ้าระบุ NOT NULL หรือ CONSTANT keyword ในการประกาศตัวแปรแสดงว่า ต้องมีการกำหนด Initial value ให้ตัวแปรนั้นด้วย

1) การประกาศตัวแปรแบบทั่วไป

```
SET SERVEROUTPUT ON
DECLARE
Myname VARCHAR2(20):= 'John';
BEGIN
Myname := 'Steven';
DBMS_OUTPUT.PUT_LINE('My name is: '||Myname);
END;
/
```

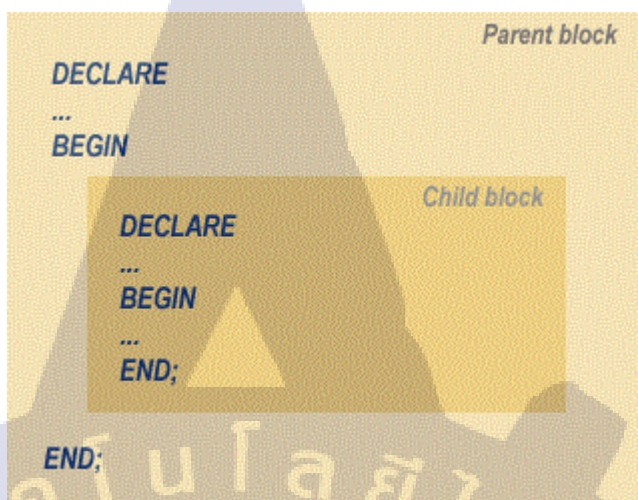
2) การประกาศตัวแปรแบบกำหนดค่าโดยที่ค่านั้นมี 's อยู่ในข้อความ

```
SET SERVEROUTPUT ON
DECLARE
V_event VARCHAR2(20);
BEGIN
V_event := q[Father's Day];
DBMS_OUTPUT.PUT_LINE('The event is: '|| V_event);
END;
/
```

3) การประกาศใช้งานตัวแปรแบบ Substitution Variable
ซึ่งเป็นรูปแบบที่ให้ผู้ใช้เป็นผู้ป้อนค่าให้กับตัวแปร

```
SET SERVEROUTPUT ON
DECLARE
Service_charge NUMBER(2):= &service_charge;
BEGIN
DBMS_OUTPUT.PUT_LINE('Service charge is: '||service_charge);
END;
/
```

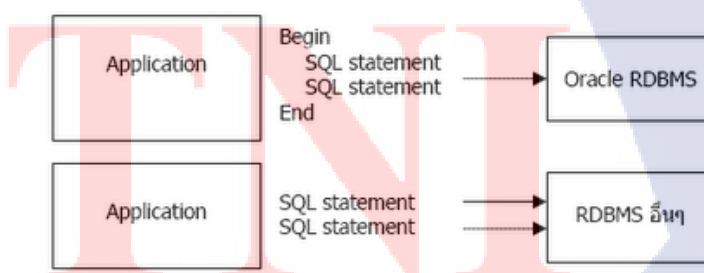
การใช้งาน Nested Block



ภาพที่ 2.2 โครงสร้างการเขียน Block ใน PL/SQL แบบ Nested Block

ประสิทธิภาพของ PL/SQL

ถ้าไม่มี PL/SQL, RDBMS จะประมวลผล SQL Statement ครั้งละ 1 Statement ทำให้ประมวลผลล่าช้า ถ้าทำงานบน Network ที่มีผู้ใช้หลายคนสั่ง Run SQL Statement พร้อมกันในเวลาเดียวกันจะทำให้เพิ่ม Traffic บน Network ทำงานเพิ่มขึ้น ถ้ามี PL/SQL เราส่ง SQL statement ในลักษณะของ Block (มีหลาย SQL Statements ในหนึ่ง Block) ไปประมวลผลที่ RDBMS Engine วิธีนี้ทำให้ลดการติดต่อสื่อสารระหว่างโปรแกรมที่พัฒนากับ RDBMS และลด Traffic บน Network ไปในตัว



ภาพที่ 2.3 เปรียบเทียบการส่ง PL/SQL และ SQL Statement

จากภาพที่ 2.3 เป็นการเปรียบเทียบการส่ง SQL Statement ไปประมวลผลระหว่าง PL/SQL ของ Oracle กับ SQL ของโปรดัคส์อื่น Oracle ส่งไปเป็น Block ใน Block ประกอบด้วยหลาย SQL statements เมื่อตัวจัดการฐานข้อมูล (RDBMS) ได้รับ SQL Code ที่มาเป็น

Block ทำให้การประมวลผลทำได้เร็วกว่ารับมาครั้งละ Statement แล้วประมวลผลวิธีนี้ทำให้ PL/SQL เพิ่มประสิทธิภาพการประมวลของ RDBMS

2.4.1 การใช้งานคำสั่ง SQL ใน PL/SQL

■ ตัวอย่าง SELECT

```
SET SERVEROUTPUT ON
DECLARE
sum_sal NUMBER(10,2);
deptno  NUMBER NOT NULL := 60;
BEGIN
SELECT SUM(salary) -- group function
INTO sum_sal FROM employees
WHERE department_id = deptno;
DBMS_OUTPUT.PUT_LINE ('The sum of salary is '|| sum_sal);
END;
/
```

%TYPE and %ROWTYPE

- %TYPE

ดึงชนิดข้อมูลจากคอลัมน์ที่กำหนดจากตารางมาเก็บไว้

ตัวอย่างการใช้งาน

```
DECLARE
your_varemployees.employee_id%TYPE;
BEGIN
for v_record in (select * from employees)
loop
your_var:=v_record.employee_id;
dbms_output.put_line(your_var);
end loop;
END;
```

- %ROWTYPE

ดึงชนิดข้อมูลจากทุกคอลัมน์ในตารางมาเก็บไว้

ตัวอย่างการใช้งาน

DECLARE

your_varemployees%ROWTYPE;

BEGIN

for v_record in (select * from employees)

loop

your_var:=v_record;

dbms_output.put_line(your_var.employee_id);

end loop;

END;

■ INSERT/UPDATE/DELETE

BEGIN

INSERT INTO employees

(employee_id, first_name, last_name, email, hire_date, job_id, salary)

VALUES(employees_seq.NEXTVAL, 'Ruth', 'Cores', 'RCORES',sysdate,

'AD_ASST', 4000);

END;

/

2.4.2 Cursor ใน PL/SQL

Cursor เป็นโครงสร้างข้อมูลของ PL/SQL หรือชนิดข้อมูล pointer ที่อ้างอิงถึง work area เมื่อมีการรันคำสั่ง SQL ในฐานข้อมูล Oracle จะมีการสร้างพื้นที่ในหน่วยความจำภายในฐานข้อมูลเพื่อใช้ในการทำงานกับคำสั่ง SQL

Cursor แบ่งออกเป็นสองประเภทดังนี้

Implicit cursors :

คือ Cursor ที่ถูกสร้างโดยอัตโนมัติ เมื่อมีการเรียกใช้คำสั่ง DML เช่น INSERT, UPDATE, และ DELETE และ SELECT ที่ส่งคืนค่าเพียงรายการเดียว

- SQL%FOUND จะเป็นTRUE ก็ต่อเมื่อมีมากกว่า 1 record ที่เป็นผลจากการทำงานของคำสั่ง SQL ล่าสุด
- SQL%NOTFOUND จะเป็นTRUE ก็ต่อเมื่อไม่มี record ใดเลยที่เป็นผลจากการทำงานของคำสั่ง SQL ล่าสุด
- SQL%ROWCOUNTจะแสดงจำนวน record ทั้งหมดที่เป็นผลจากการทำงานของคำสั่ง SQL ล่าสุด

Explicit cursors:

คือ Cursor ที่ผู้ใช้งานสร้างขึ้นเอง ซึ่งต้องเขียนคำสั่งต่างๆ เพื่อมาจัดการ Cursor เอง

2.4.3 คำสั่งตรวจสอบเงื่อนไข**IF...ELSE**

```
IF condition THEN statements;
[ELSIF condition THEN statements;]
[ELSE statements;]
END IF;
```

CASE...WHEN

```
CASE selector
WHEN expression1 THEN result1
WHEN expression2 THEN result2
...
WHEN expressionN THEN resultN
[ELSE resultN+1]
END;
```

2.4.4 คำสั่งควบคุมการวนซ้ำ

Basic loop

LOOP

statement1;

...

EXIT [WHEN condition];

END LOOP;

FOR loop

FOR counter IN [REVERSE]

lower_bound..upper_bound LOOP

statement1;

...

END LOOP;

WHILE loop

WHILE *condition* LOOP

statement1;

statement2;

...

END LOOP;

2.4.5 การสร้าง Function

```
CREATE [OR REPLACE] FUNCTION function_name
[(parameter1 [mode1] datatype1, ...)]
RETURN datatype IS|AS
[local_variable_declarations; ...]
BEGIN
  -- actions;
  RETURN expression;
END [function_name];
```

ภาพที่ 2.4 การสร้าง Function

2.4.6 การสร้าง Procedure

```
CREATE [OR REPLACE] PROCEDURE procedure_name
[(parameter1 [mode] datatype1,
  parameter2 [mode] datatype2, ...)]
IS|AS
[local_variable_declarations; ...]
BEGIN
  -- actions;
END [procedure_name];
```

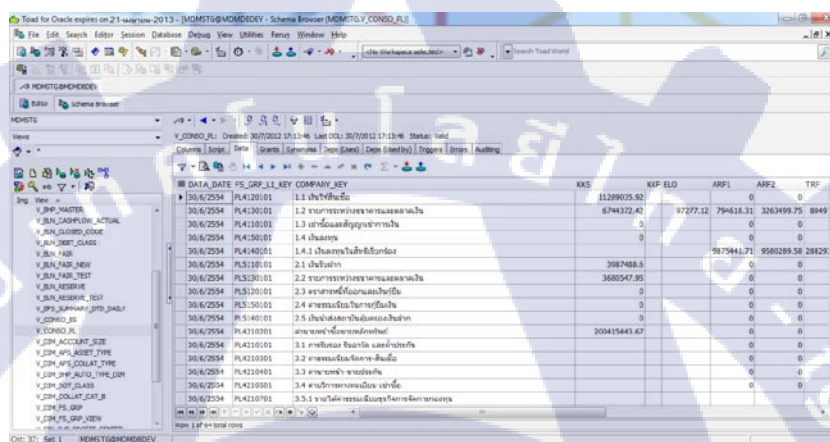
ภาพที่ 2.5 การสร้าง Procedure

2.5 Jdeveloper

Jdeveloper เป็น Tools ของทาง Oracle ให้บริการในการสร้าง Application หรือ BPEL ซึ่ง Tools แต่ละตัวจะให้บริการต่างกันโดย การติดต่อกับ Database สามารถใช้วิธีติดต่อแบบ Connection Pool คือการติดต่อกับ Database เพียงครั้งเดียวสามารถเรียกใช้เมื่อไหร่ก็ได้โดยจะถูกเรียกใช้ในรูปแบบของ Class โดยส่วนใหญ่จะสร้างแบบ สถาปัตยกรรม MVC เพื่อให้เก็บข้อมูลไว้ที่ Model และสามารถเรียกใช้เมื่อไหร่ก็ได้ ถ้าเป็นของ BPEL การทำ Connection Pool ทำให้เมื่อต้องการเรียก Data Adapter จะอ้างอิงจาก Connection เพื่อให้ Partner Link สามารถติดต่อกับ Database ได้และให้บริการการออกแบบชนิด GUI ของ BPEL และ Wizard ที่รองรับการสร้าง Schema เมื่อเรียก Partner Link โดยอัตโนมัติทำให้สะดวกต่อการพัฒนา

2.6 TOAD For Oracle

TOAD For Oracle เป็นเครื่องมือที่ช่วยอำนวยความสะดวกในการเข้าไปติดต่อและจัดการฐานข้อมูลของ Oracle เมื่อเราต้องการใช้คำสั่ง SQL กับ Oracle เราอาจจะใช้ SQL*Plus เพื่อ run คำสั่ง SQL และเมื่อเราต้องการ manage ตัว Database แบบ GUI ก็อาจจะใช้ Enterprise Manager เป็นต้น แต่ TOAD สามารถทำงานทั้งสองอย่างนี้ได้ในตัวมันเอง และยังมี feature อื่นๆอีกมากมายให้ใช้งาน ที่สำคัญคือใช้งานง่าย



DATA	DATE	FS	SRP	LL	KEY	COMPANY	KEY	KKS	KXP	ELO	ARF1	ARF2	TRF
30/6/2554	PL4120101	1.1	เช่าที่ดิน					11289035.92			0	0	
30/6/2554	PL4130101	1.2	เช่าที่ดิน					6744322.42	97277.12	754618.31	3263499.75	8849	
30/6/2554	PL4130101	1.3	เช่าที่ดิน					0	0	0	0	0	
30/6/2554	PL4130101	1.4	เช่าที่ดิน					0	0	0	0	0	
30/6/2554	PL4140101	1.4.1	เช่าที่ดิน					0	0	0	0	0	
30/6/2554	PL5110101	2.1	เช่าที่ดิน					9875441.71	9580289.59	28428			
30/6/2554	PL5130101	2.2	เช่าที่ดิน					3680547.95	0	0	0	0	
30/6/2554	PL5130101	2.3	เช่าที่ดิน					0	0	0	0	0	
30/6/2554	PL5130101	2.4	เช่าที่ดิน					0	0	0	0	0	
30/6/2554	PL5140101	2.5	เช่าที่ดิน					0	0	0	0	0	
30/6/2554	PL4210101	3.1	เช่าที่ดิน					200415443.47	0	0	0	0	
30/6/2554	PL4210101	3.2	เช่าที่ดิน					0	0	0	0	0	
30/6/2554	PL4210101	3.3	เช่าที่ดิน					0	0	0	0	0	
30/6/2554	PL4210101	3.4	เช่าที่ดิน					0	0	0	0	0	
30/6/2554	PL4210101	3.5	เช่าที่ดิน					0	0	0	0	0	

ภาพที่ 2.6 แสดงตัวอย่างโปรแกรม TOAD For Oracle

บทที่ 3

แผนงานการปฏิบัติงานและขั้นตอนการดำเนินงาน

3.1 แผนงานปฏิบัติงาน

ตารางที่ 3.1 แผนปฏิบัติงานโครงการ

หัวข้องาน	เดือนที่ 1	เดือนที่ 2	เดือนที่ 3	เดือนที่ 4
1.ศึกษา Business ของลูกค้า				
2.ศึกษา Tool ที่ใช้งาน				
3.พัฒนา Web Application Project Pursuit CR 25				
4.ทดสอบระบบ และนำไปใช้งานจริง				
5.Research JavaScript เพื่อใช้สร้าง PDF From				
6.ทำ ADF Style Template				

3.2 รายละเอียดโครงการ

ข้าพเจ้าได้เข้ามาสหกิจศึกษา ณ บริษัท A-Host Company Limited แผนก Academy of Advanced Service เป็นระยะเวลา 4 เดือน มีหน้าที่ที่ได้รับมอบหมายในการทำงานเกี่ยวกับระบบงานเร่งรัดของธนาคารเกียรตินาคิน โดยเริ่มจากการศึกษาโค้ดและโปรแกรมเพื่อพัฒนาแอปพลิเคชันตามความต้องการของลูกค้า (Change Request 25) ซึ่งนับเป็นงานหลักเพราะใช้เวลาในการศึกษาและพัฒนามากกว่าหัวข้องานอื่นๆ

3.3 ขั้นตอนการดำเนินงานที่นักศึกษาปฏิบัติงานหรือโครงการ

ขั้นตอนการดำเนินงานแบ่งออกเป็น 3 ส่วน ได้แก่ การศึกษาโครงการ ความต้องการของลูกค้า(Change Request 25) และการทดสอบระบบ

3.3.1 การศึกษาโครงการ

ลักษณะงานที่ได้รับมอบหมาย คือ การศึกษาการทำงานของระบบ HPAOL และระบบ Pursuit เดิมจากเอกสารที่ได้รับ นอกจากนั้นยังได้รับมอบหมายให้ทำระบบจำลองจากระบบจริงที่ใช้อยู่โดยใช้เครื่องที่บริษัทเป็นserverแทน ซึ่งในserverจะมีข้อมูลจำลองจากเครื่องธนาคาร เพื่อศึกษาตัวระบบให้เข้าใจและสามารถพัฒนาระบบให้สามารถทดสอบหรือใช้งานจริงได้ โดยจะอธิบายขั้นตอนการใช้งาน Oracle JDeveloper เพื่อจำลองหน้าค้นหาข้อมูล และแก้ไขข้อมูลตามที่ระบบจริงใช้งานอยู่เนื่องจากข้อมูลของธนาคารไม่สามารถเปิดเผยได้

3.3.1.1 เริ่มใช้งานโปรแกรมOracle Jdeveloper

a) ทำการแตกไฟล์โปรแกรมลงบนเครื่องดังภาพ

Name	Date modified	Type	Size
 jdevstudio10133_whSVN.zip	12/2/2009 10:08 AM	เอกสาร WinRAR แ...	511,104 KB

ภาพที่ 3.1 ขั้นตอนการเริ่มใช้งาน ขั้นตอนที่ 1

b) ดับเบิ้ลคลิก jdeveloper.exe เพื่อเปิดโปรแกรม ดังภาพ

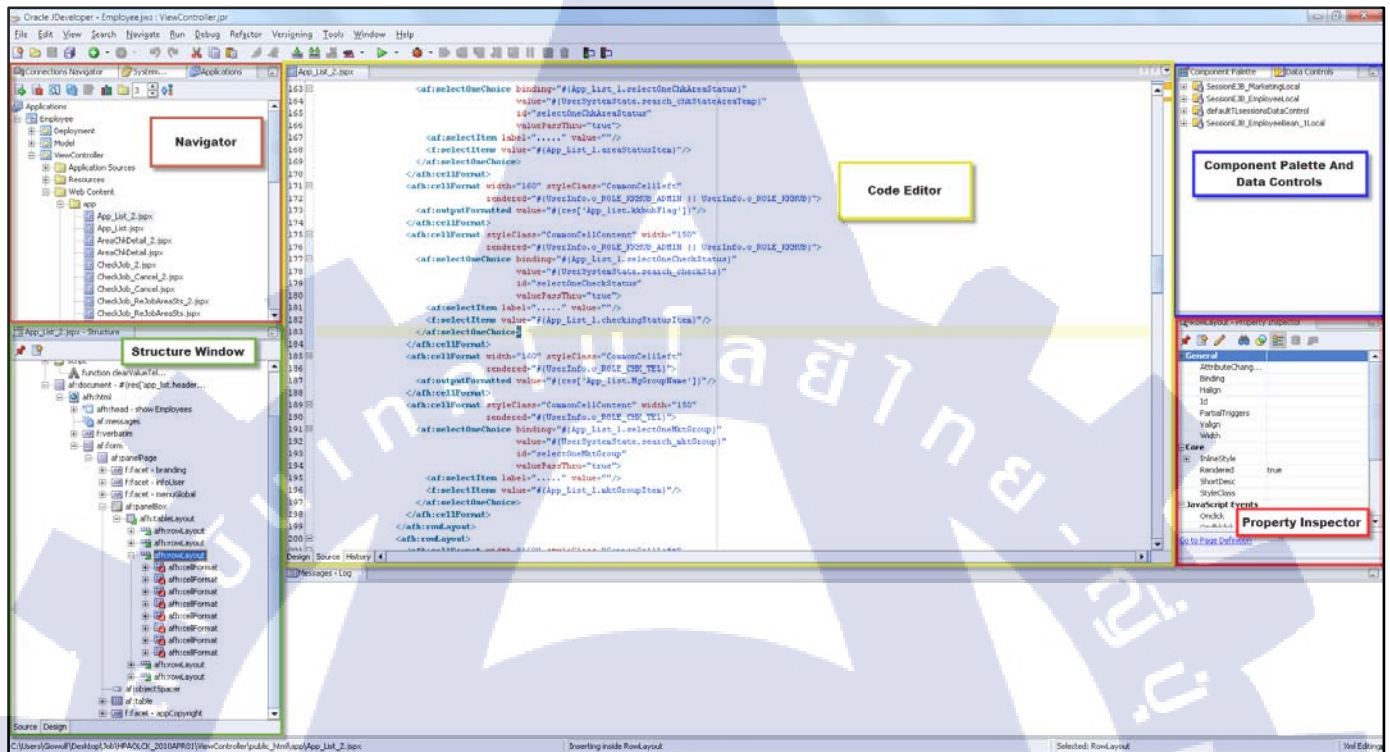
	xqs	19-Jun-07 3:28 PM	File folder	
	jdeveloper.exe	19-Jun-07 1:17 AM	Application	29 KB
	jdeveloper.exe	Date modified: 19-Jun-07 1:17 AM	Date created: 05-Aug-13 4:38 PM	Size: 28.5 KB

ภาพที่ 3.2 ขั้นตอนการเริ่มใช้งาน ขั้นตอนที่ 2

3.3.1.2 ตั้งค่าOracleJdeveloper ก่อนเริ่มสร้างงาน

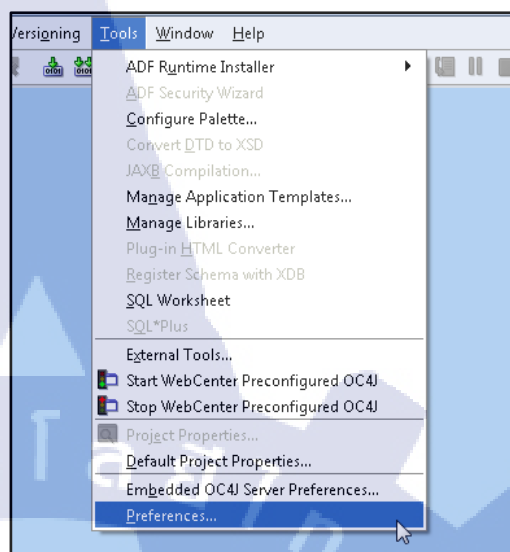
ก่อนอื่นข้าพเจ้าจะอธิบายส่วนประกอบต่างๆ ของโปรแกรม

Jdeveloper



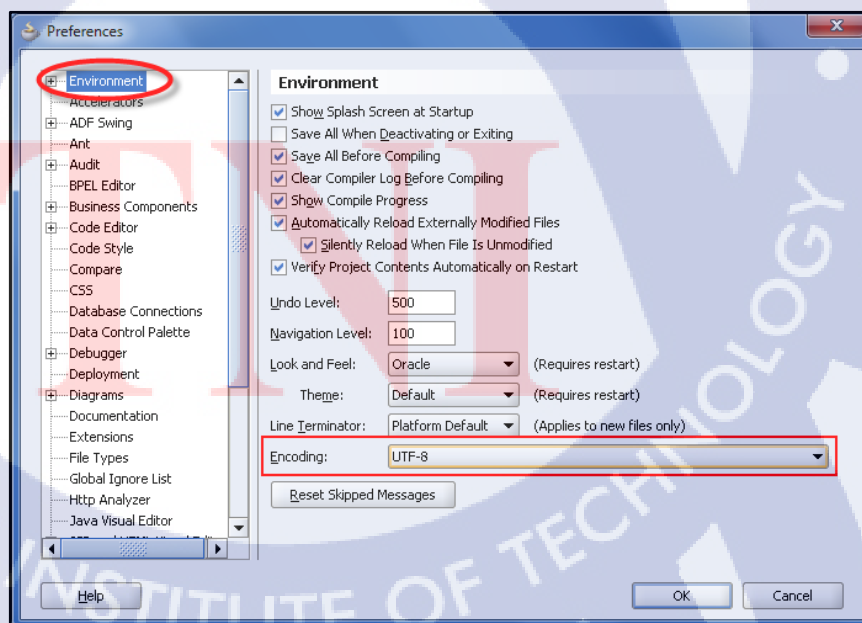
ภาพที่ 3.3 ส่วนประกอบต่างๆ ของโปรแกรม Jdeveloper

a) ไปยังแถบเครื่องมือ Tools เลือก Preferences....ดังภาพ



ภาพที่ 3.4 ขั้นตอนการตั้งค่า ขั้นตอนที่ 1

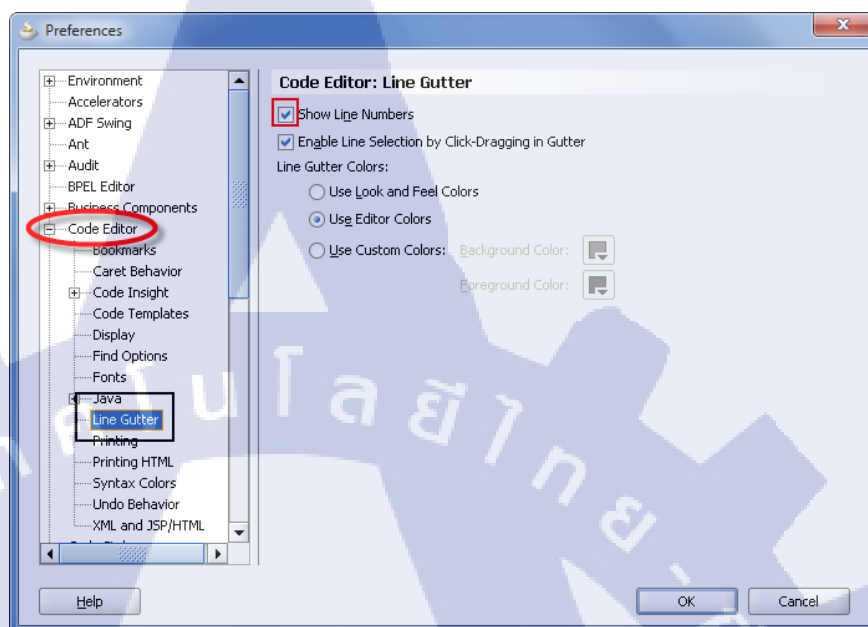
b) เลือก Environment แก้ Encoding ให้เป็น UTF-8ดังภาพ(เพื่อให้สามารถอ่านค่าภาษาไทยได้)



ภาพที่

3.5 ขั้นตอนการตั้งค่า ขั้นตอนที่ 2

c) เลือก Code Editor → Line Gutter ทำการเลือก show Line Numbers
 ดังภาพ (เพื่อช่วยในการเช็ค error เพราะการแจ้ง error จะแจ้ง line number ออกมา)

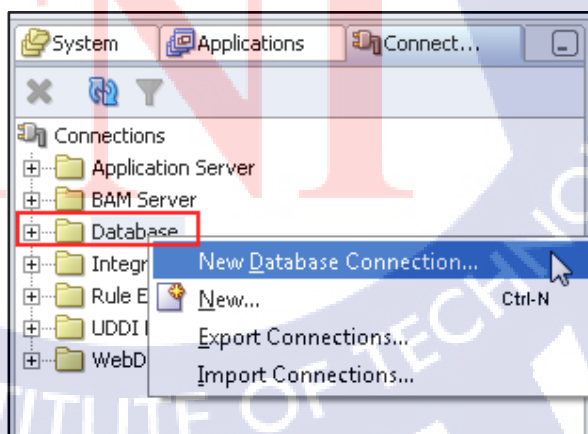


ภาพที่

3.6 ขั้นตอนการตั้งค่า ขั้นตอนที่ 3

3.3.1.3 สร้างการเชื่อมต่อฐานข้อมูล

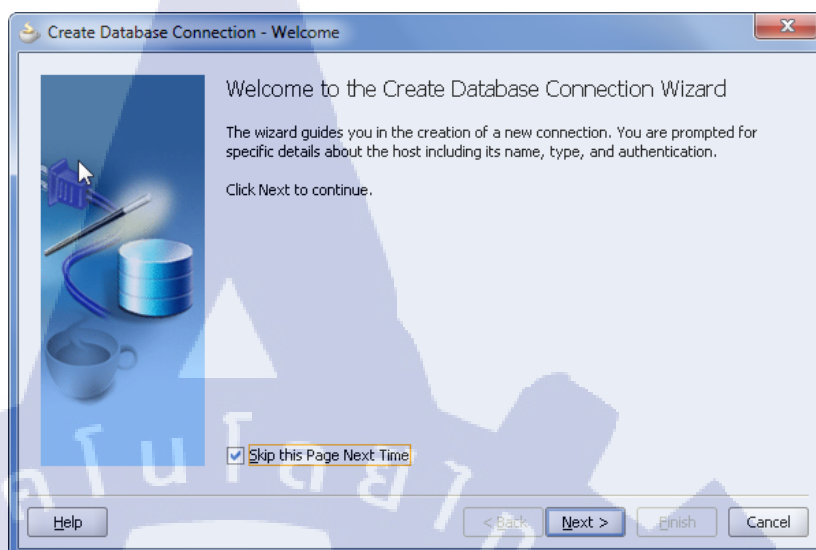
a) ไปยังแถบ Navigator เลือก connections คลิกขวา ที่ โฟลเดอร์
 Database เลือก New Database Connection... ดังภาพ



ภาพที่

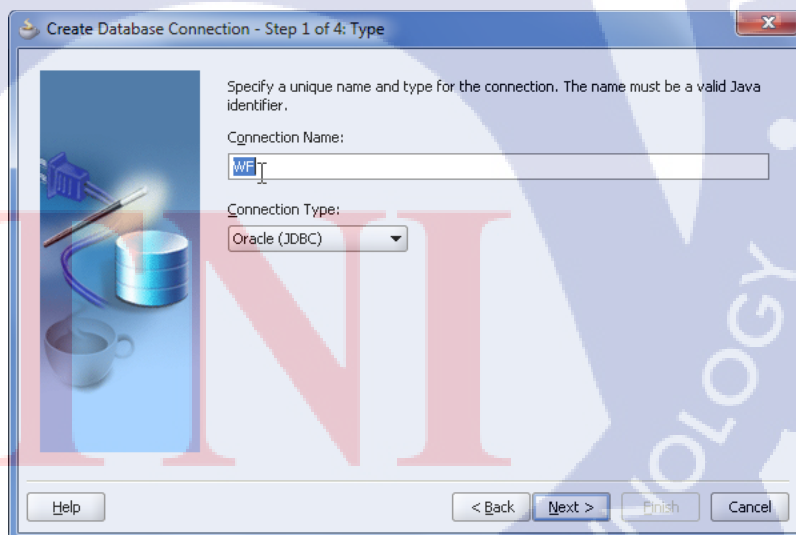
3.7 ขั้นตอนการเชื่อมต่อฐานข้อมูล ขั้นตอนที่ 1

b) คลิก Next บนหน้า Welcome ดังภาพ



ภาพที่ 3.8 ขั้นตอนการเชื่อมต่อฐานข้อมูล ขั้นตอนที่ 2

c) ตั้งชื่อ Connection Name ว่า WF จากนั้นคลิก Next ดังภาพ



ภาพที่ 3.9 ขั้นตอนการเชื่อมต่อฐานข้อมูล ขั้นตอนที่ 3

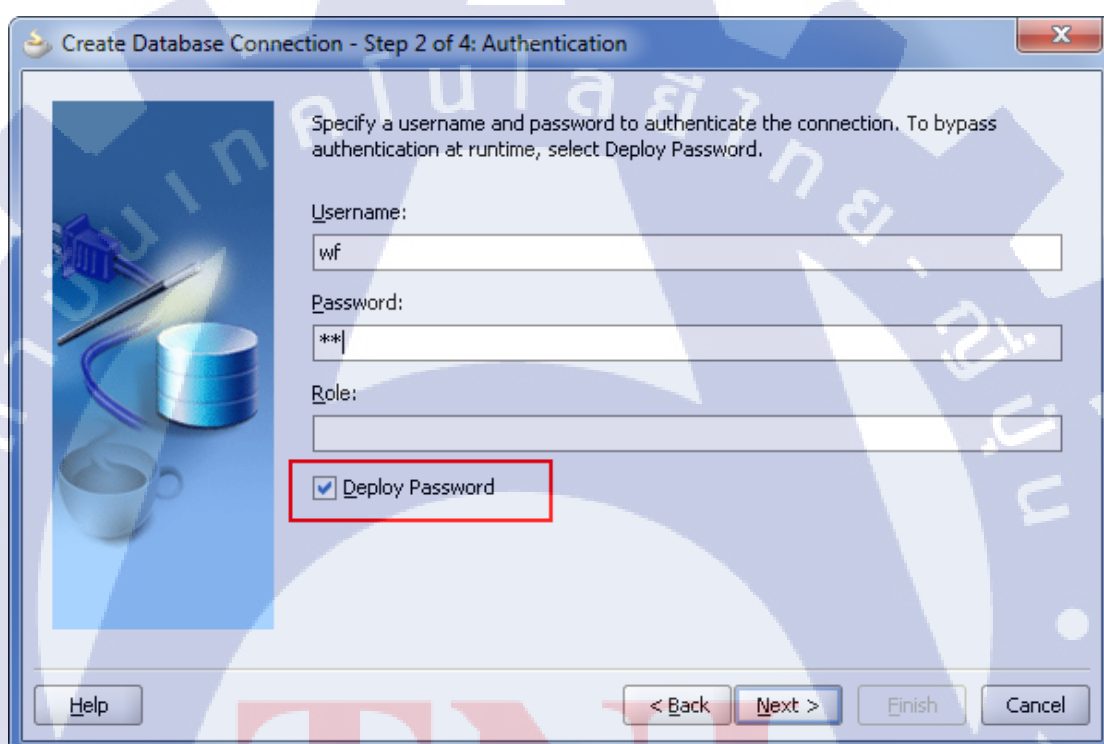
d) ในหน้า Authentication ใส่ข้อมูลดังนี้

Username: wf

Password: wf

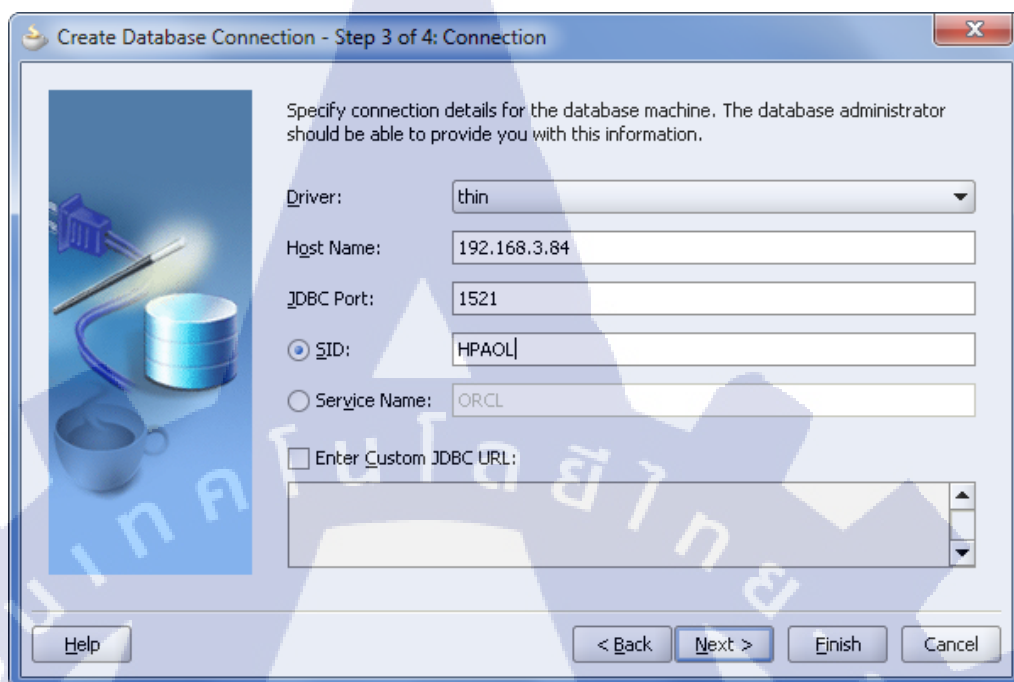
Deploy Password: ทำการเลือก

จากนั้นคลิก Next ดังภาพ



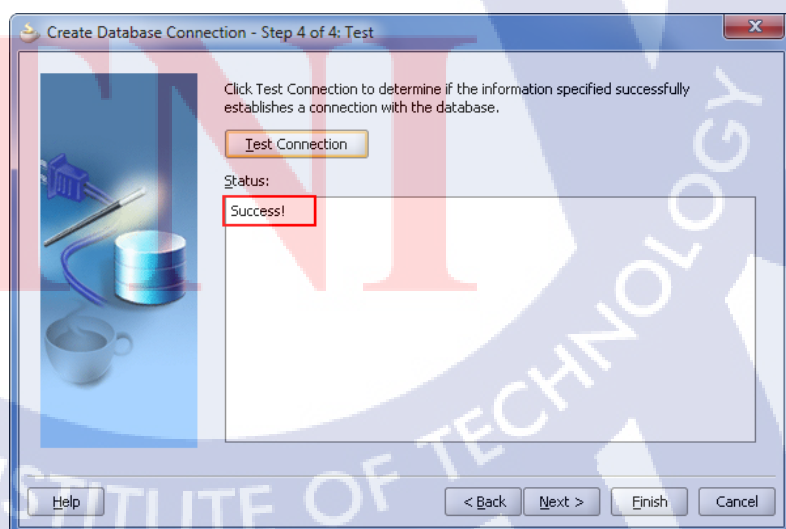
ภาพที่ 3.10 ขั้นตอนการเชื่อมต่อฐานข้อมูล ขั้นตอนที่ 4

e) ในหน้า Connection กรอกข้อมูลดังภาพ จากนั้นคลิก Next ดังภาพ



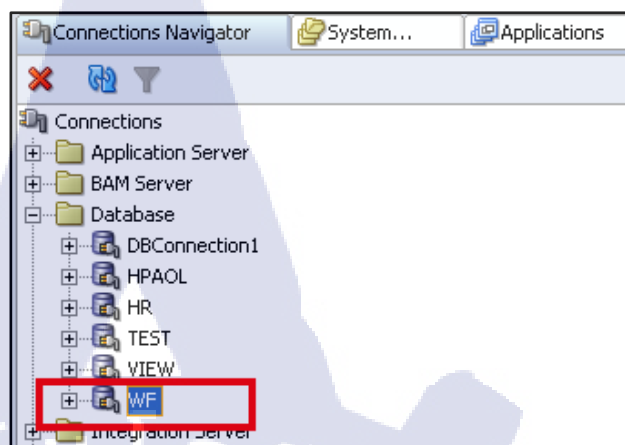
ภาพที่ 3.11 ขั้นตอนการเชื่อมต่อฐานข้อมูล ขั้นตอนที่ 5

f) คลิก Test Connection เพื่อทำการทดสอบการเชื่อมต่อ เมื่อปรากฏคำว่า **Success!** หมายความว่า การเชื่อมต่อฐานข้อมูลสำเร็จ จากนั้นคลิก Finish ดังภาพ



ภาพที่ 3.12 ขั้นตอนการเชื่อมต่อฐานข้อมูล ขั้นตอนที่ 6

g) การเชื่อมต่อที่สร้างขึ้นจะมาปรากฏอยู่ในโฟลเดอร์ Database ดังภาพ

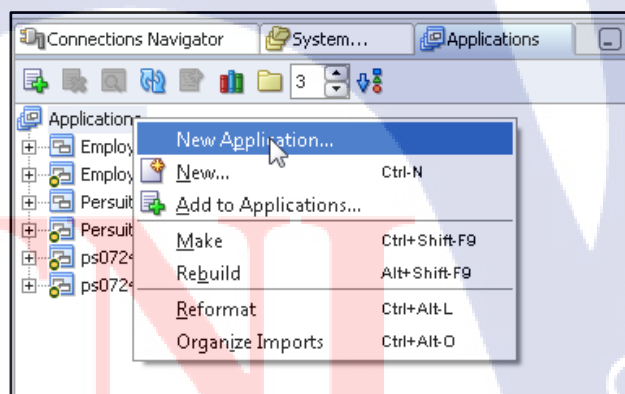


ภาพที่

3.13 ขั้นตอนการเชื่อมต่อฐานข้อมูล ขั้นตอนที่ 7

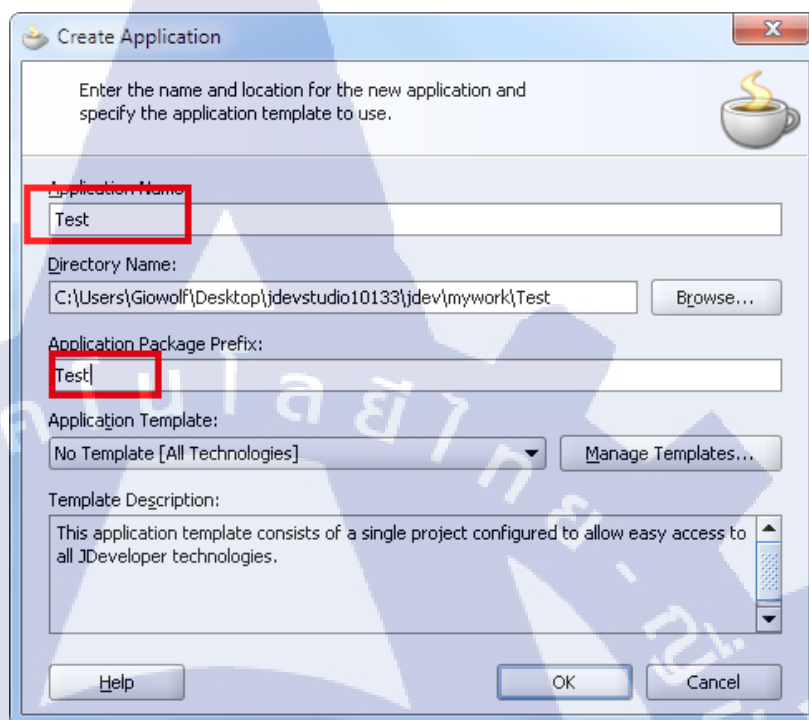
3.3.1.4 สร้าง Application & Model Project

a) คลิกขวาที่ Application ในแถบ Navigator เลือก New Application..ดังภาพ



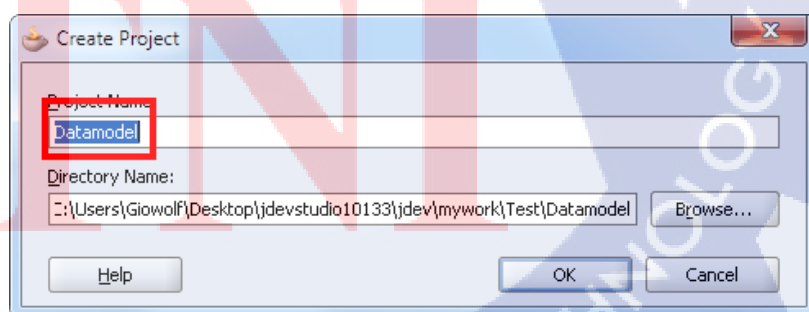
ภาพที่ 3.14 ขั้นตอนการสร้าง Application & Model Project ขั้นตอนที่ 1

b) ตั้งชื่อ Application Name: **Test** และ Application Package Prefix: **test**
จากนั้น คลิก OK ดังภาพ (โปรแกรมจะบังคับให้สร้าง Project ดังข้อถัดไป)



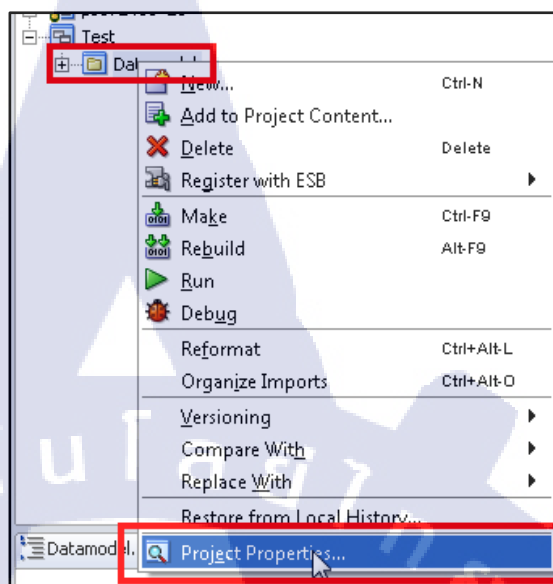
ภาพที่ 3.15 ขั้นตอนการสร้าง Application & Model Project ขั้นตอนที่ 2

c) ตั้งชื่อ Project Name: **Data Model** จากนั้นคลิก OK ดังภาพ



ภาพที่ 3.16 ขั้นตอนการสร้าง Application & Model Project ขั้นตอนที่ 3

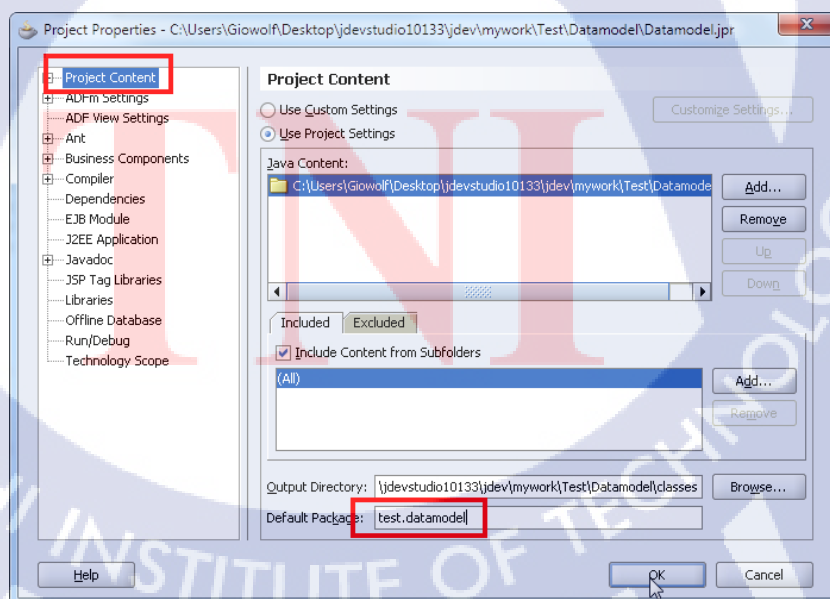
d) คลิกขวาที่ DataModel ในแถบ Navigator เลือก Project Properties ดังภาพ



ภาพที่ 3.17 ขั้นตอนการสร้าง Application & Model Project ขั้นตอนที่ 4

e) เลือก Project Content ตั้งชื่อ Default Package: **test.datamodel**

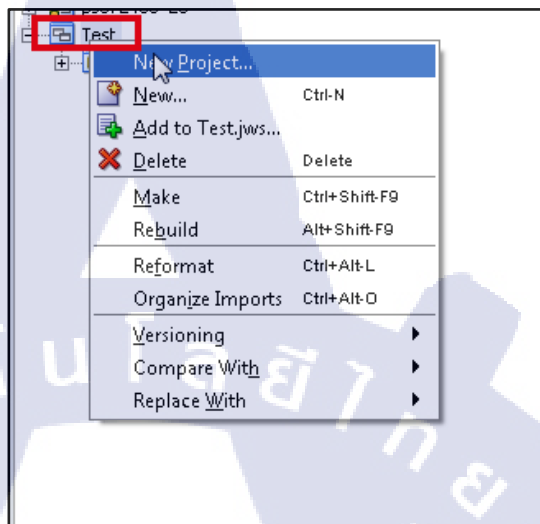
จากนั้นคลิก OK ดังภาพ



ภาพที่ 3.18 ขั้นตอนการสร้าง Application & Model Project ขั้นตอนที่ 5

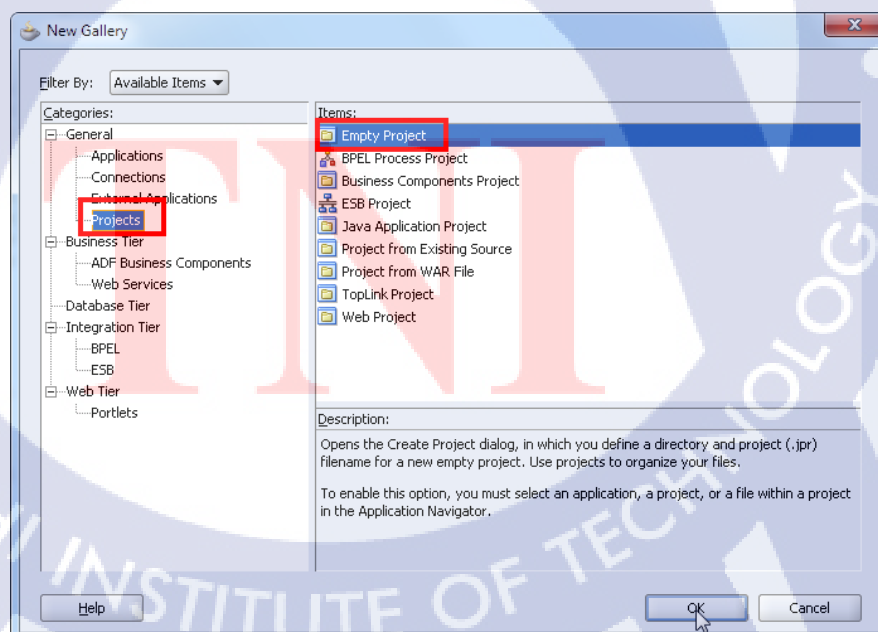
3.3.1.5 สร้าง View & Controller Project

a) คลิกขวาที่ Test ในแถบ Navigator เลือก New Project.. ดังภาพ



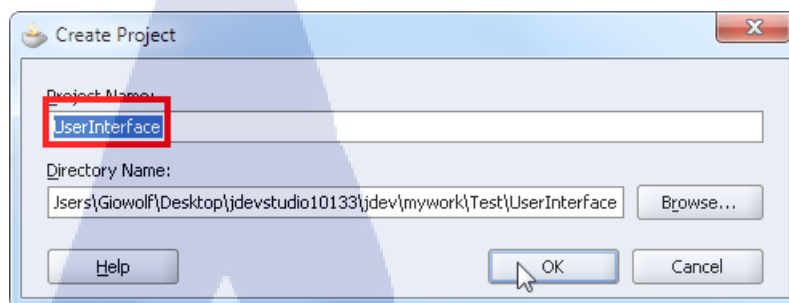
ภาพที่ 3.19 ขั้นตอนการสร้าง View& Controller ขั้นตอนที่ 1

b) เลือก Projects → Empty Project จากนั้นคลิก OK ดังภาพ



ภาพที่ 3.20 ขั้นตอนการสร้าง View& Controller ขั้นตอนที่ 2

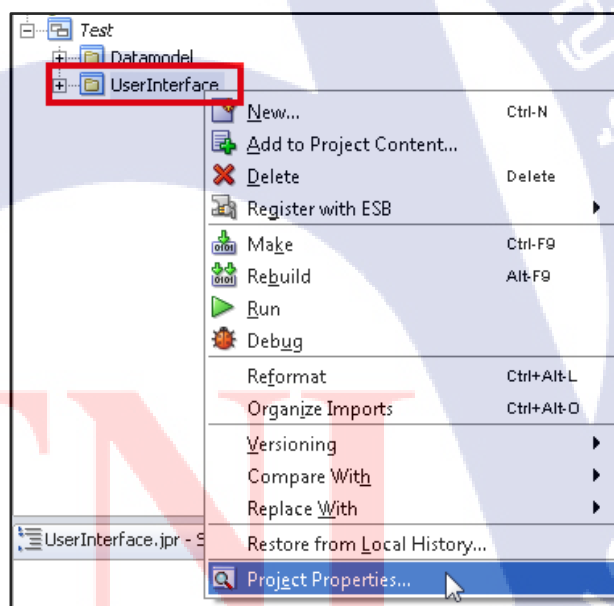
c) ตั้งชื่อ Project Name: **UserInterface** จากนั้นคลิก OK ดังภาพ



ภาพที่ 3.21 ขั้นตอนการสร้าง View& Controller ขั้นตอนที่ 3

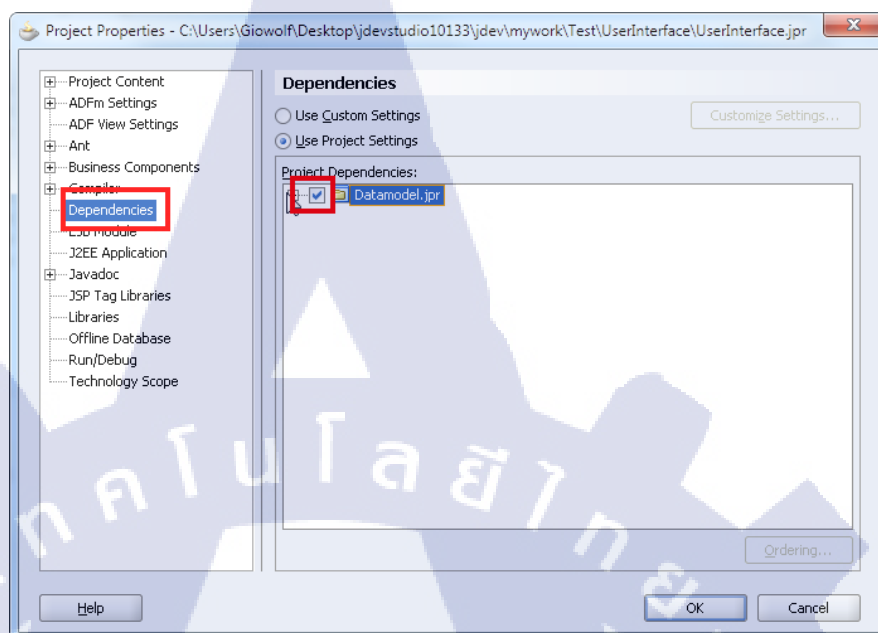
d) คลิกขวาที่ UserInterface ในแถบ Navigator เลือก Project Properties..

ดังภาพ



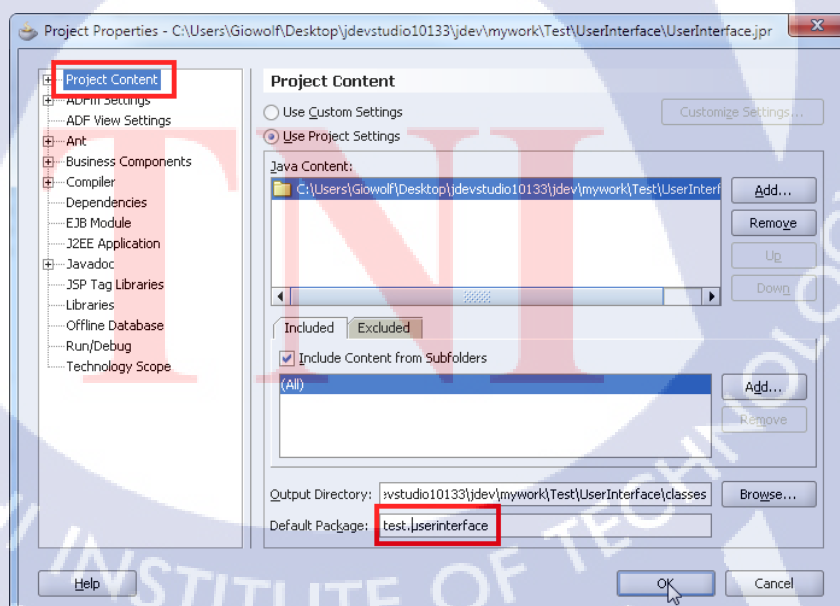
ภาพที่ 3.22 ขั้นตอนการสร้าง View&Controller ขั้นตอนที่ 4

e) ไปยัง Dependencies ทำการเลือก checkboxDataModel.jpr ดังภาพ



ภาพที่ 3.23 ขั้นตอนการสร้าง View&Controller ขั้นตอนที่ 5

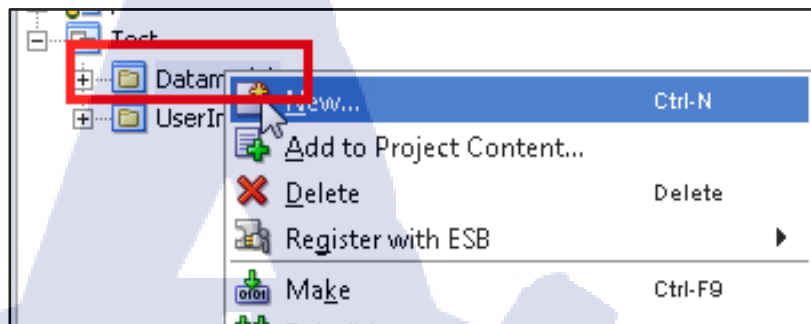
f) ไปยัง Project Content ตั้งชื่อ Default Package: test.userinterface ดังภาพ



ภาพที่ 3.24 ขั้นตอนการสร้าง View&Controller ขั้นตอนที่ 6

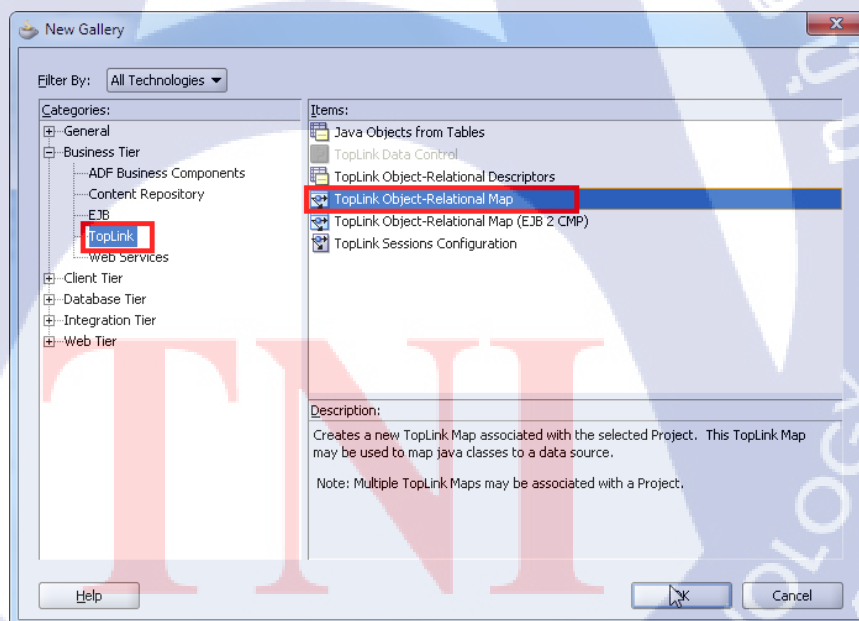
3.3.1.6 สร้าง Toplink Map

a) คลิกขวาที่ DataModel ในแถบ Navigator เลือก New... ดังภาพ



ภาพที่ 3.25 ขั้นตอนการสร้าง Toplink Map ขั้นตอนที่ 1

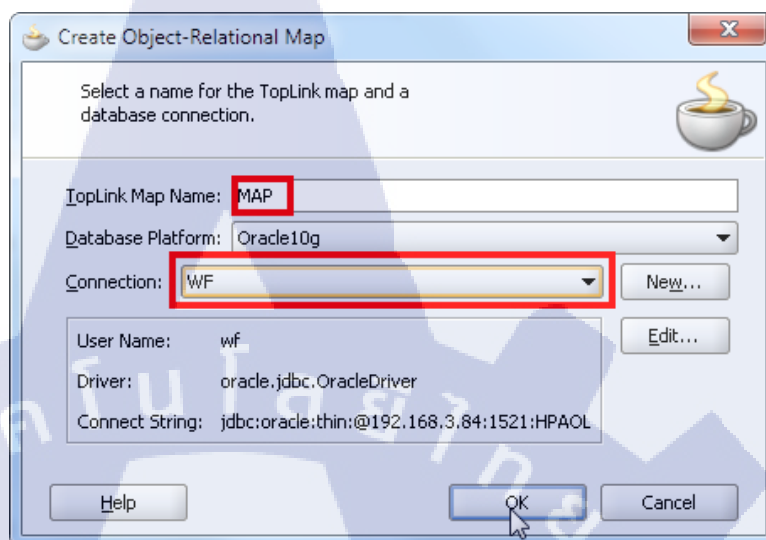
b) TopLink → TopLink Object-Relational Map จากนั้นคลิก OK ดังภาพ



ภาพที่ 3.26 ขั้นตอนการสร้าง Toplink Map ขั้นตอนที่ 2

c) ตั้งชื่อ TopLink Map Name: **MAP** / Connection: **WF** จากนั้นคลิก OK

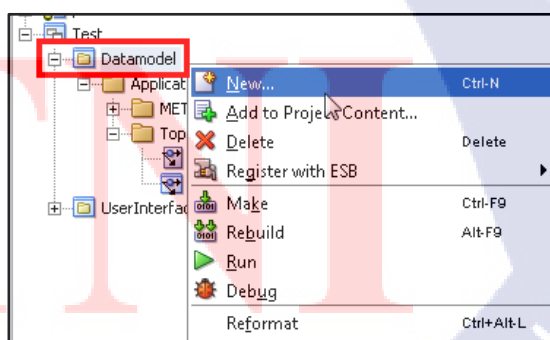
ดั่งภาพ



ภาพที่ 3.27 ขั้นตอนการสร้าง Toplink Map ขั้นตอนที่ 3

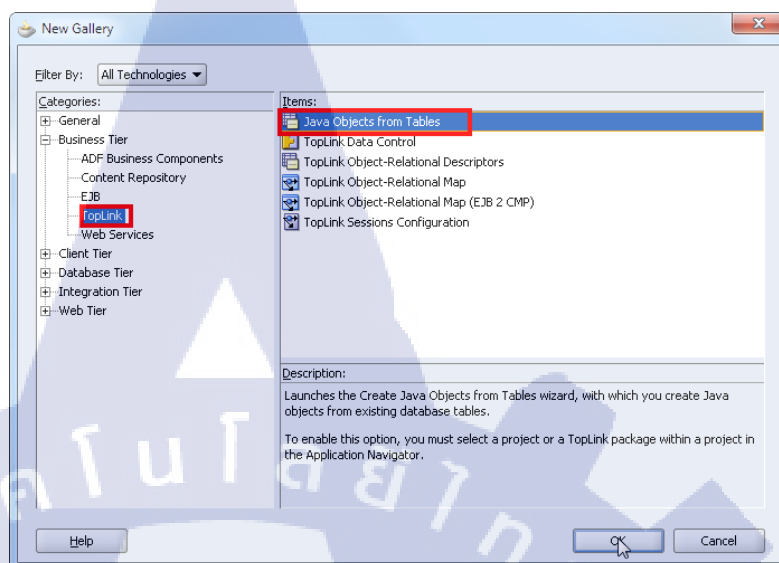
3.3.1.7 สร้าง Data Model Object

a) คลิกขวา Data Model เลือก New.. ดั่งภาพ



ภาพที่ 3.28 ขั้นตอนการสร้าง Data Model Object ขั้นตอนที่ 1

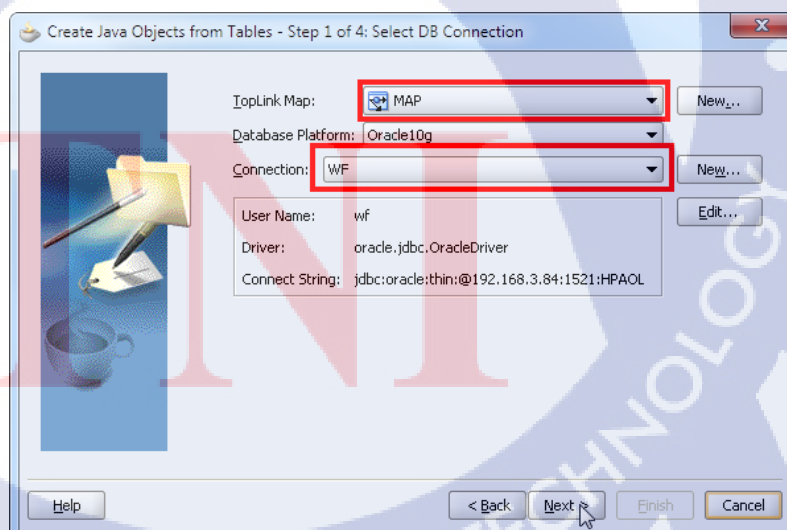
b) TopLink → Java Objects from Tables จากนั้นคลิก OK ดังภาพ



ภาพที่ 3.29 ขั้นตอนการสร้าง Data Model Object ขั้นตอนที่ 2

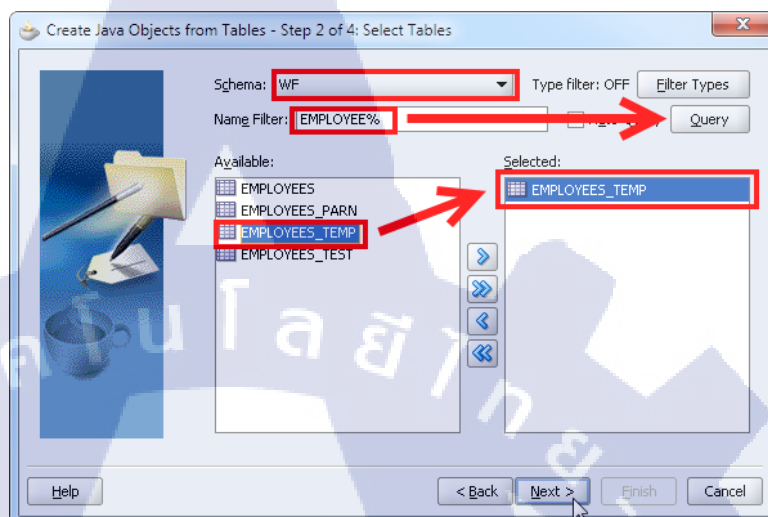
c) เลือก TopLink Map: **MAP** / Connection: **WF** จากนั้นคลิก Next ดัง

ภาพ



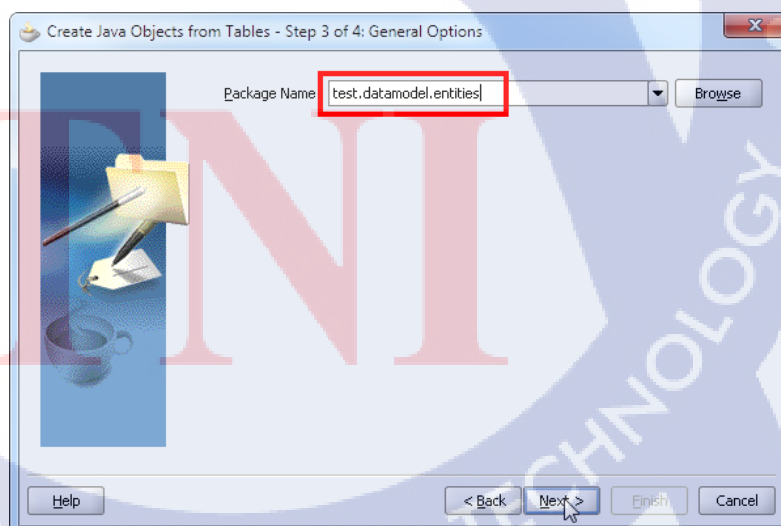
ภาพที่ 3.30 ขั้นตอนการสร้าง Data Model Object ขั้นตอนที่ 3

d) Name Filter พิมพ์ EMPLOYEE% จากนั้นคลิก Query เพื่อแสดงตารางที่มีคำว่า EMP เท่านั้น เลือก EMPLOYEES_TEMP คลิก> เพื่อเลือกตาราง (ตารางจะมาอยู่ด้านขวา) จากนั้นคลิก Next ดังภาพ



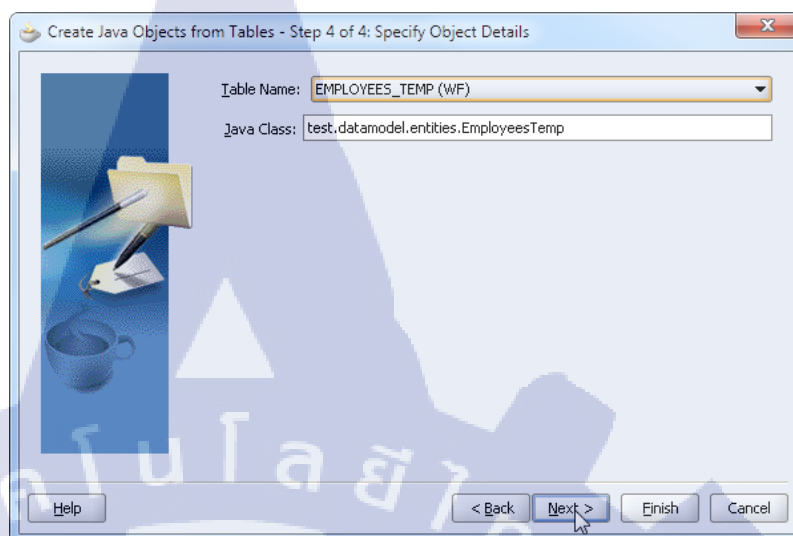
ภาพที่ 3.31 ขั้นตอนการสร้าง Data Model Object ขั้นตอนที่ 4

e) Package Name: **test.datamodel.entities** จากนั้นคลิก Next ดังภาพ



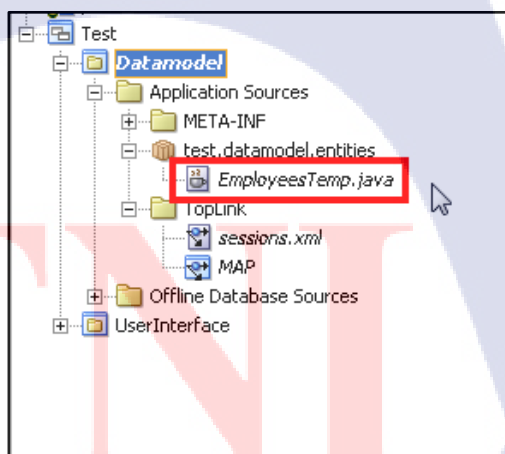
ภาพที่ 3.32 ขั้นตอนการสร้าง Data Model Object ขั้นตอนที่ 5

f) Table Name: EMPLOYEES_TEMP(WF) จากนั้นคลิก Next ดังภาพ



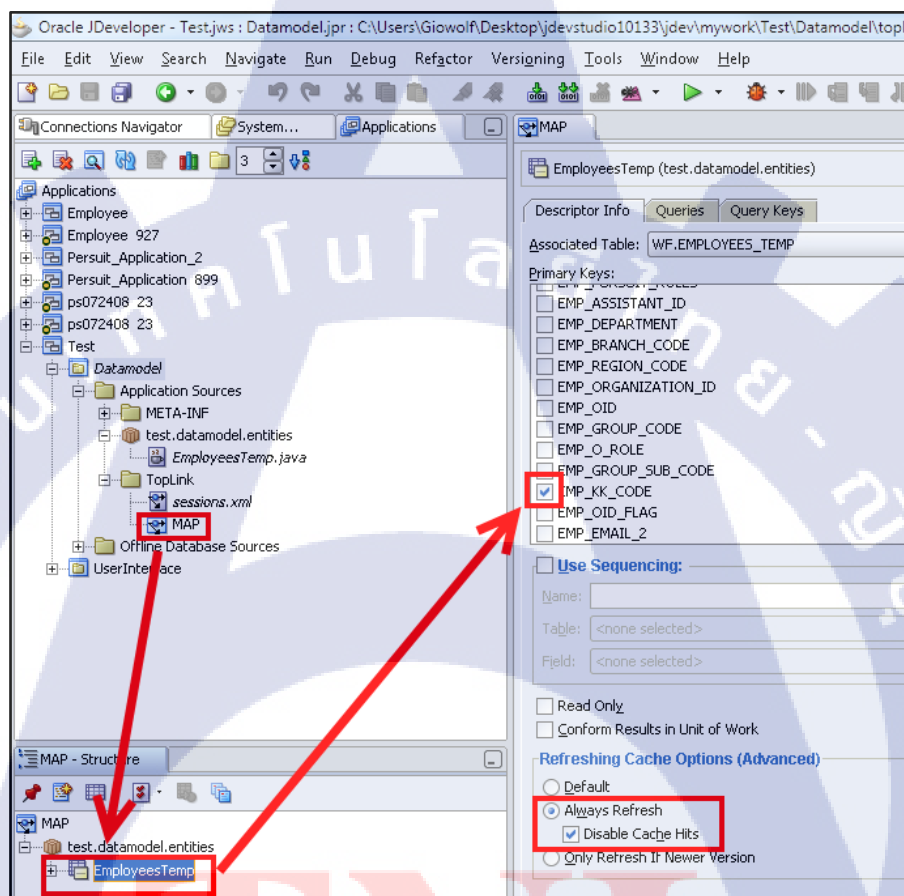
ภาพที่ 3.33 ขั้นตอนการสร้าง Data Model Object ขั้นตอนที่ 6

g) จะปรากฏไฟล์ดังภาพ



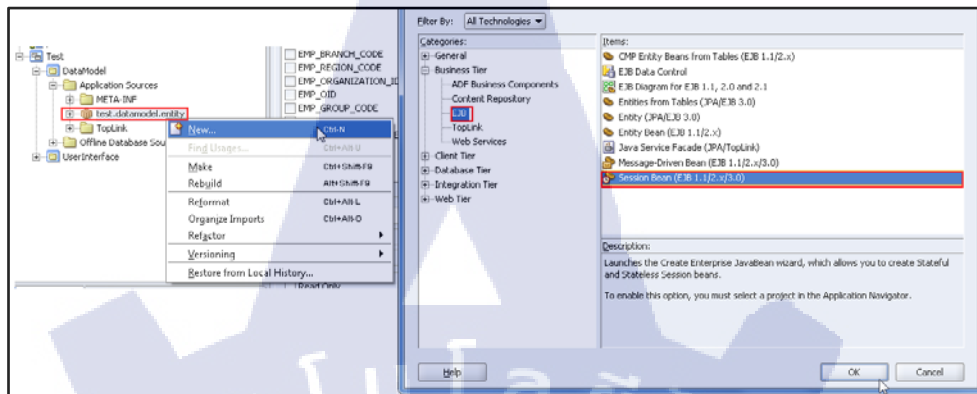
ภาพที่ 3.34 ขั้นตอนการสร้าง Data Model Object ขั้นตอนที่ 7

h) คลิก MAP จากปรากฏ EmployeesTemp ใน Structure window ดับเบิ้ลคลิก EmployeesTemp จะปรากฏหน้าต่าง MAP ในด้านซ้ายมือจากนั้นตั้ง Primary Keys เป็น **EMP_KK_CODE** ต่อมาส่วนของ Refreshing Cache Options(Advanced) เลือก Radio button เป็น Always Refresh และทำการเลือก Check box Disable Cache Hits ดังภาพ



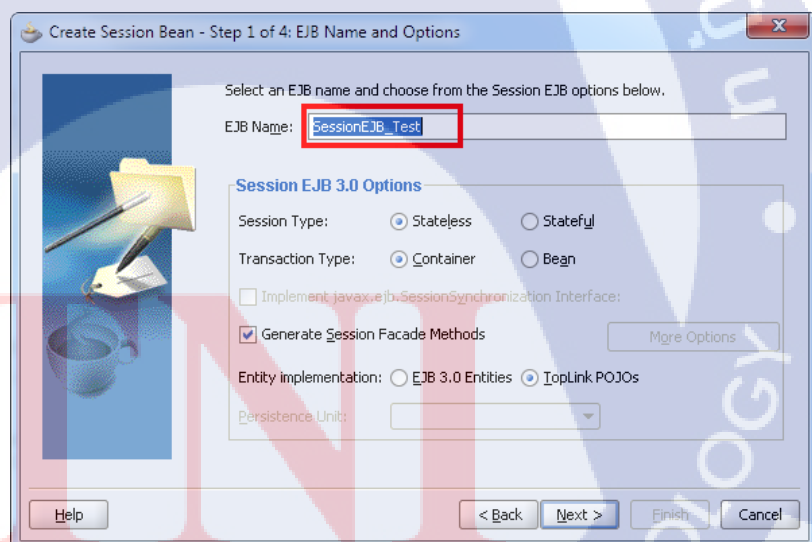
ภาพที่ 3.35 ขั้นตอนการสร้าง Data Model Object ขั้นตอนที่ 8

- i) คลิกขวาที่ test.datamodel.entity ในแถบ Navigator เลือก New.. ไปที่ EJB → Session Bean (EJB 1.1/2.x/3.0) จากนั้นคลิก OK ดังภาพ



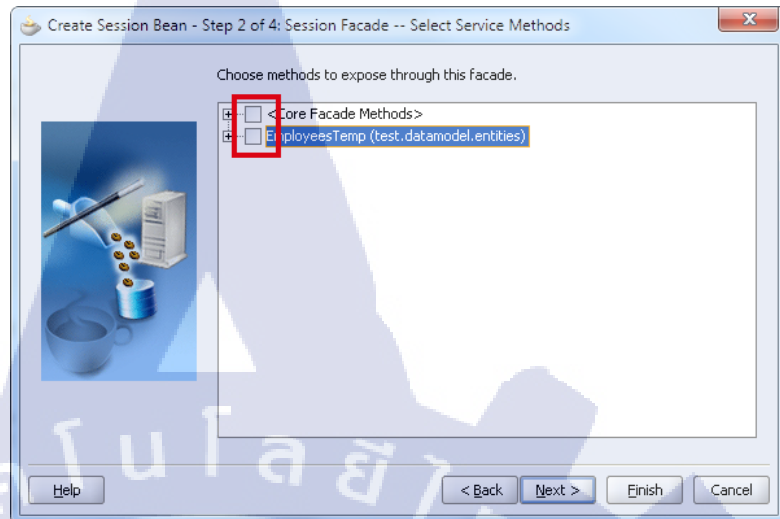
ภาพที่ 3.36 ขั้นตอนการสร้าง Data Model Object ขั้นตอนที่ 9

- j) ตั้งชื่อ EJB Name: SessionEJB_Test จากนั้นคลิก Next ดังภาพ



ภาพที่ 3.37 ขั้นตอนการสร้าง Data Model Object ขั้นตอนที่ 10

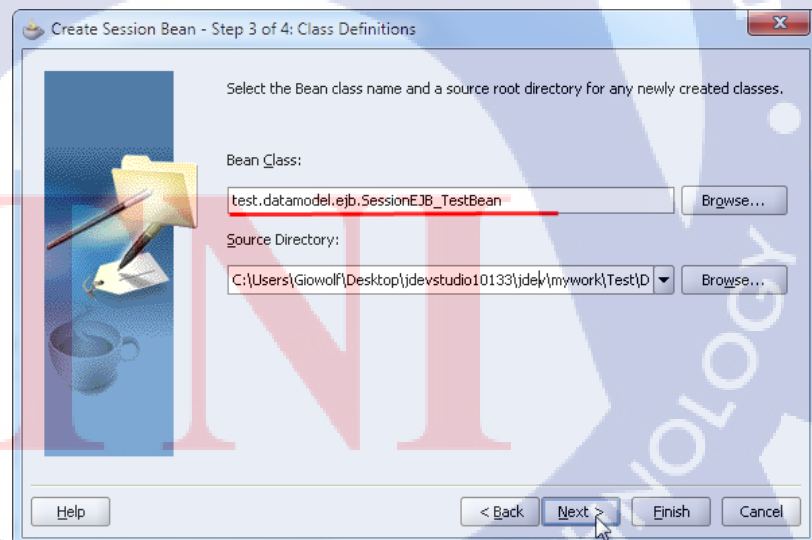
k) ไม่ทำการเลือก Check box ใดๆเลย จากนั้นคลิก Next ดังภาพ



ภาพที่ 3.38 ขั้นตอนการสร้าง Data Model Object ขั้นตอนที่ 11

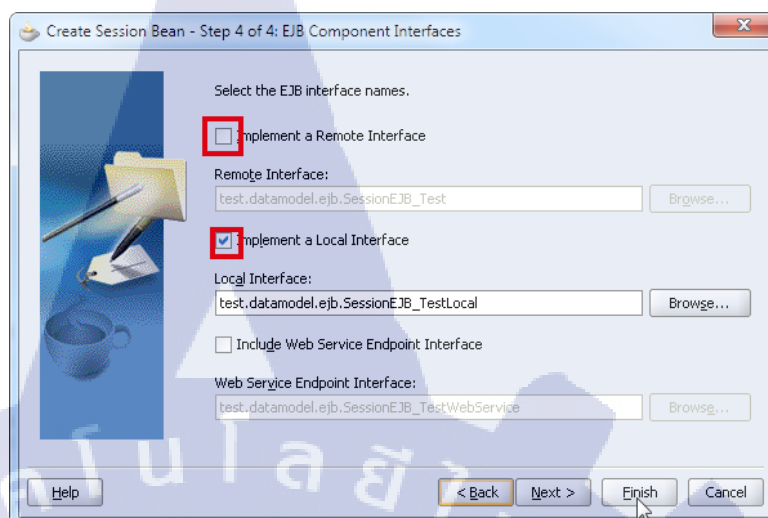
l) ตั้งชื่อ Bean Class: **test.datamodel.ejb.SessionEJB_TestBean**

จากนั้นคลิก Next ดังภาพ



ภาพที่ 3.39 ขั้นตอนการสร้าง Data Model Object ขั้นตอนที่ 12

m) ทำการเลือก Check box อันกลางเพื่ยงอันเดียวดังภาพ

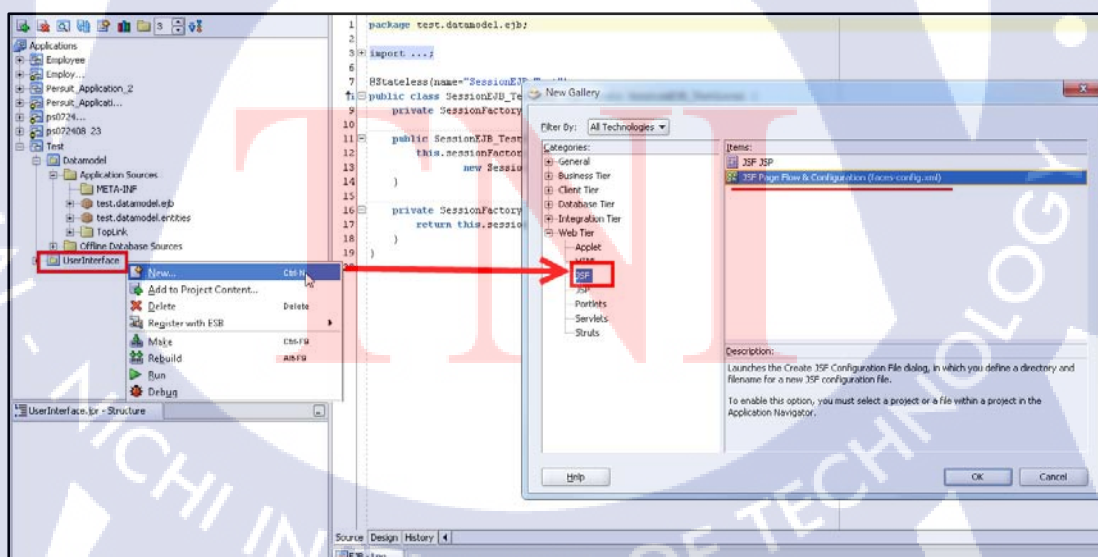


ภาพที่ 3.40 ขั้นตอนการสร้าง Data Model Object ขั้นตอนที่ 13

3.3.1.8 Configure View & Controller Project

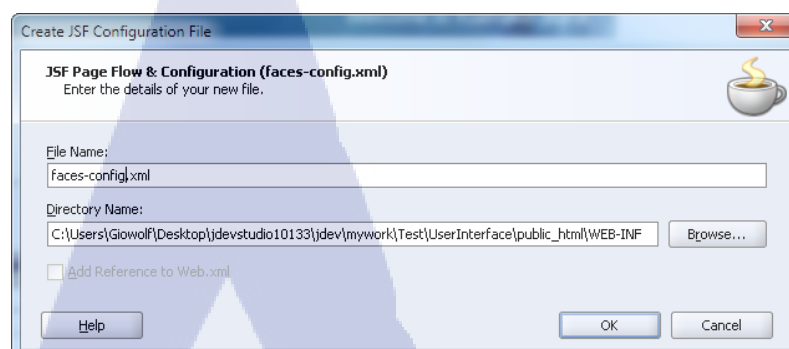
a) คลิกขวาที่ UserInterface ใน Navigator เลือก New..จากนั้นไปที่ JSF

→ JSF Page Flow & Configuration (faces-config.xml) จากนั้นคลิก OK ดังภาพ



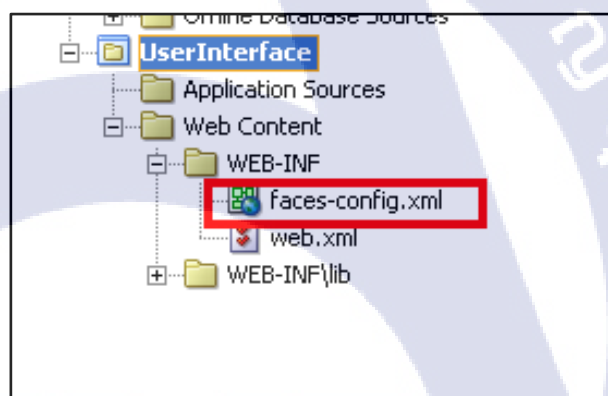
ภาพที่ 3.41 ขั้นตอนการสร้าง Configure View & Controller ขั้นตอนที่ 1

b) ตั้งชื่อ File Name: **faces-config.xml** จากนั้นคลิก OK ดังภาพ



ภาพที่ 3.42 ขั้นตอนการสร้าง Configure View & Controller ขั้นตอนที่ 2

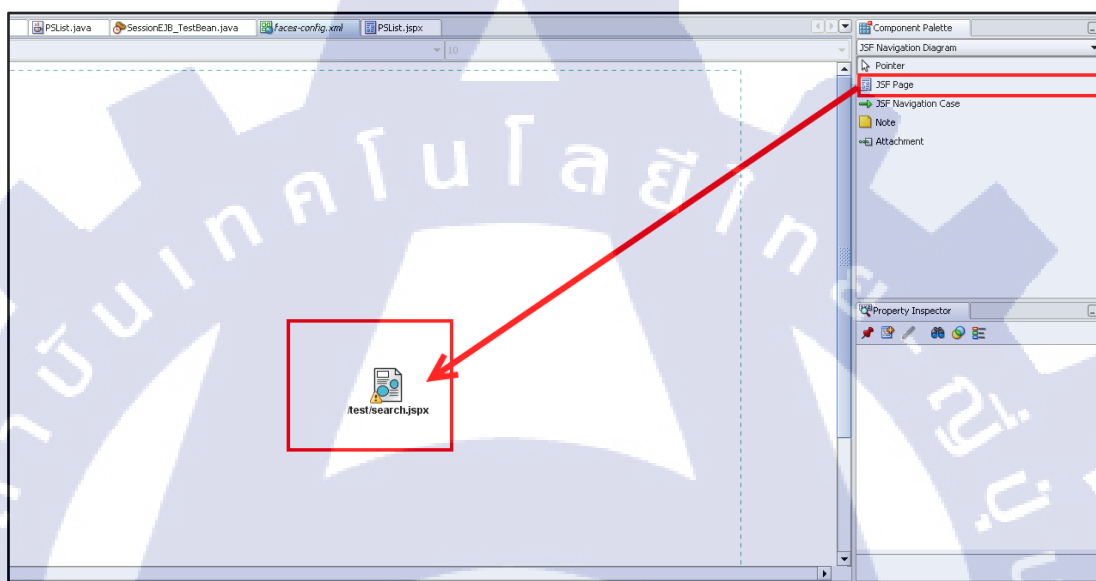
c) จะปรากฏไฟล์ faces-config.xml ดังภาพ



ภาพที่ 3.43 ขั้นตอนการสร้าง Configure View & Controller ขั้นตอนที่ 3

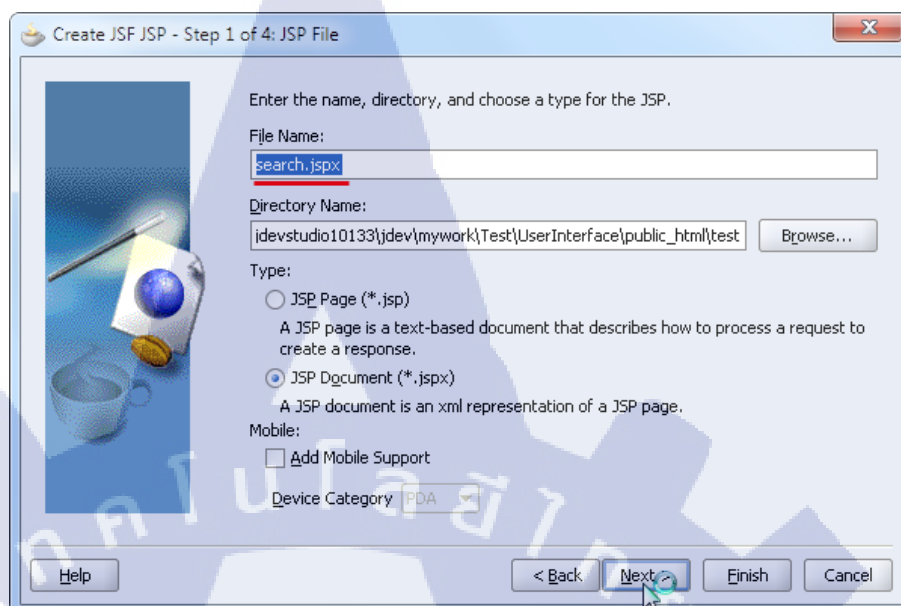
3.3.1.9 สร้างหน้าค้นหาข้อมูล

a) ดับเบิลคลิกไฟล์ faces-config.xml จะปรากฏในส่วน Code Editor
 ต่อมาไปที่ Component Palette เลือก JSF Page และคลิกในพื้นที่ว่างๆ เพื่อสร้าง object ตั้งชื่อว่า `/test/search.jspx` ที่ไอคอนจะขึ้นสัญลักษณ์อัศจรรย์ เนื่องจากหน้ายังไม่ถูกสร้าง ดังภาพ (ให้ดับเบิลคลิกคลิกที่ไอคอนดังกล่าว จะเข้าสู่ขั้นตอนการสร้างหน้า)



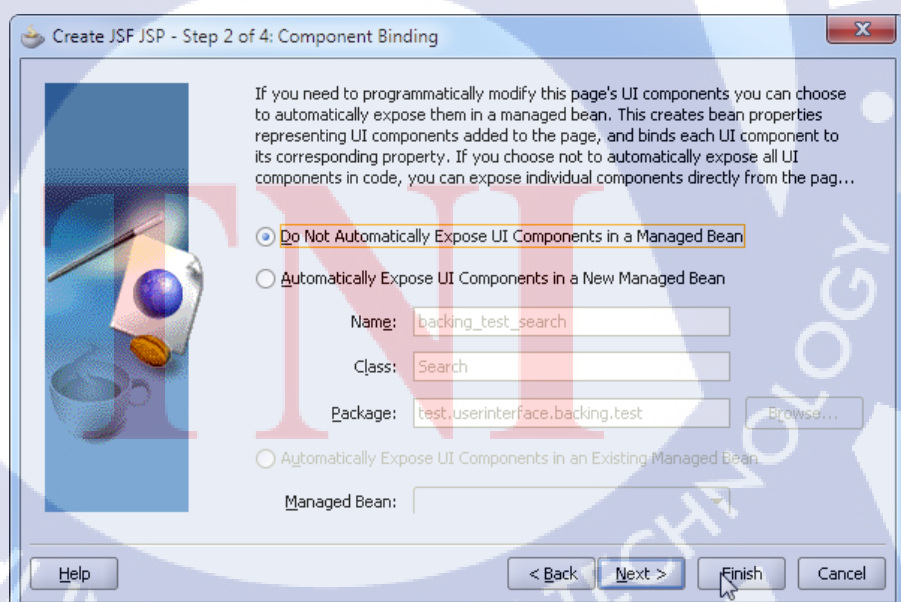
ภาพที่ 3.44 ขั้นตอนการสร้างหน้าค้นหาข้อมูล ขั้นตอนที่ 1

b) ตรวจสอบชื่อ search.jspx จากนั้นคลิก Next ดังภาพ



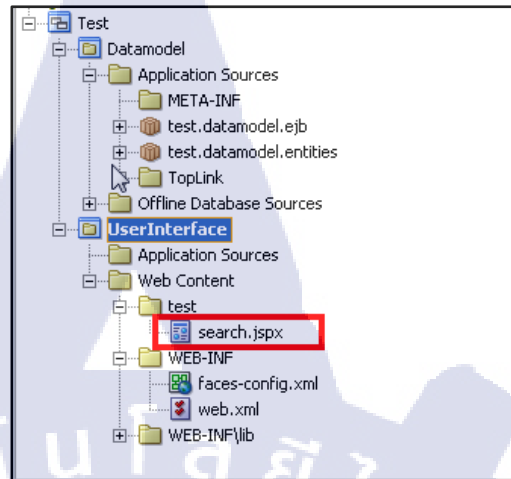
ภาพที่ 3.45 ขั้นตอนการสร้างหน้าค้นหาข้อมูล ขั้นตอนที่ 2

c) จากนั้นคลิก Finish ดังภาพ



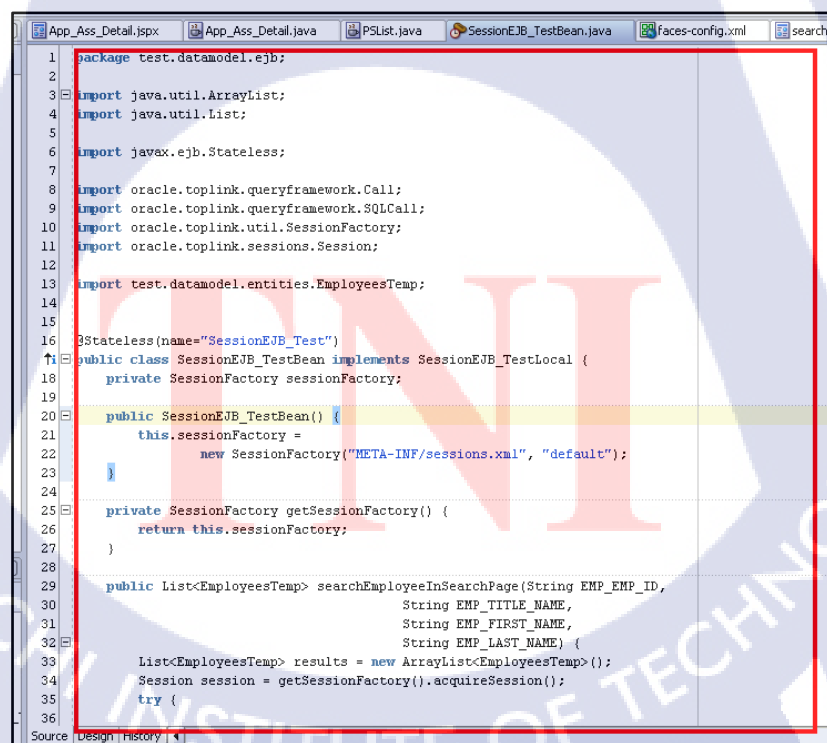
ภาพที่ 3.46 ขั้นตอนการสร้างหน้าค้นหาข้อมูล ขั้นตอนที่ 3

d) จะปรากฏไฟล์ search.jspx ดังภาพ



ภาพที่ 3.47 ขั้นตอนการสร้างหน้าค้นหาข้อมูล ขั้นตอนที่ 4

e) เปิดไฟล์ SessionEJB_TestBean.java เพื่อพิมพ์โค้ด ดังภาพ



ภาพที่ 3.48 ขั้นตอนการสร้างหน้าค้นหาข้อมูล ขั้นตอนที่ 5

โค้ดที่ถูกพิมพ์ในหน้า SessionEJB_Test Bean.java

```
package test.datamodel.ejb;

import java.util.ArrayList;

import java.util.List;

import javax.ejb.Stateless;

import oracle.toplink.queryframework.Call;
import oracle.toplink.queryframework.SQLCall;

import oracle.toplink.util.SessionFactory;

import oracle.toplink.sessions.Session;

import test.datamodel.entities.EmployeesTemp;

@Stateless(name="SessionEJB_Test")

public class SessionEJB_TestBean implements SessionEJB_TestLocal {

    private SessionFactory sessionFactory;

    public SessionEJB_TestBean() {

        this.sessionFactory =

            new SessionFactory("META-INF/sessions.xml", "default");

    }
```

```

private SessionFactory getSessionFactory() {

    return this.sessionFactory;

}

public List<EmployeesTemp> searchEmployeeInSearchPage(String EMP_EMP_ID,

    String EMP_TITLE_NAME,

    String EMP_FIRST_NAME,
    String EMP_LAST_NAME) {

    List<EmployeesTemp> results = new ArrayList<EmployeesTemp>();

    Session session = getSessionFactory().acquireSession();

    try {

        String sqlStr = "SELECT * FROM EMPLOYEES_TEMP WHERE (1=1) AND EMP_TITLE =
'MARKETING' ";

        if (EMP_EMP_ID != null && EMP_EMP_ID.length() > 0)

            sqlStr += " AND EMP_EMP_ID='" + EMP_EMP_ID + "'";

        if (EMP_TITLE_NAME != null && EMP_TITLE_NAME.length() > 0)

            sqlStr += " AND EMP_TITLE_NAME LIKE '%" + EMP_TITLE_NAME + "%'";

        if (EMP_FIRST_NAME != null && EMP_FIRST_NAME.length() > 0)

            sqlStr += " AND EMP_FIRST_NAME LIKE '%" + EMP_FIRST_NAME + "%'";

        if (EMP_LAST_NAME != null && EMP_LAST_NAME.length() > 0)

```



```
sqlStr += " AND EMP_LAST_NAME LIKE '%" + EMP_LAST_NAME + "%'";

System.out.print("\nSQL:" + sqlStr);

Call sqlCall = new SQLCall(sqlStr);

results = (List<EmployeesTemp>)session.readAllObjects(EmployeesTemp.class,

                                sqlCall);

} catch (Exception e) {

    System.err.println("(Exception Message : " + e.getMessage());

} finally {

    session.release();

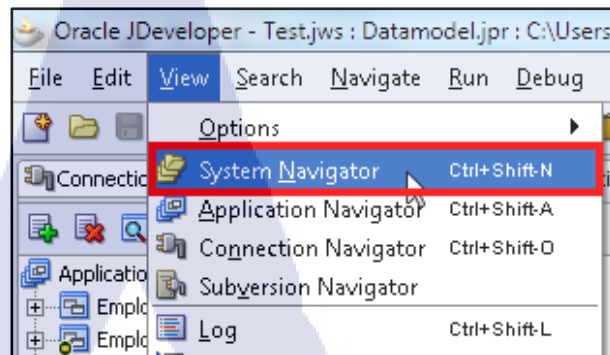
    System.out.println("Final SessionEJB_TrainingBean searchEmployee\n");

}

return results;

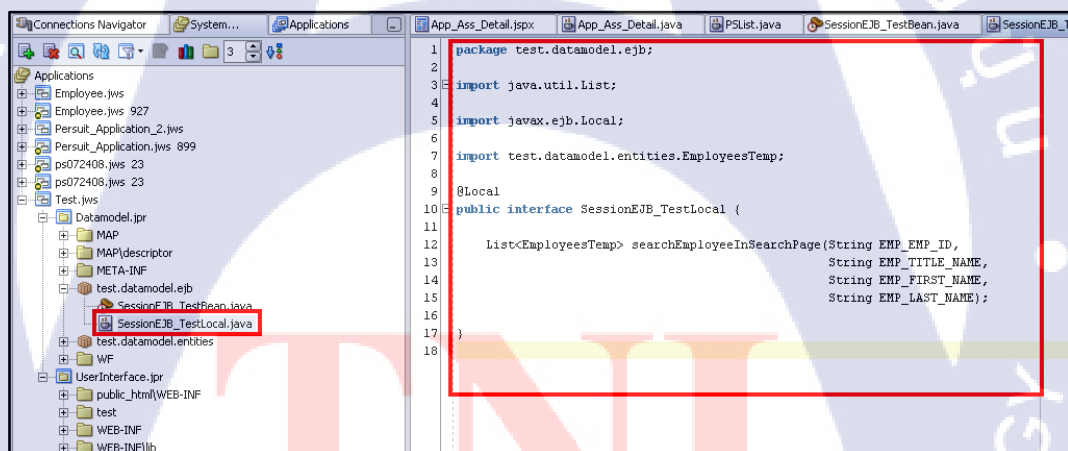
}
```

f) ไปยังเมนู View → System Navigator ดังภาพ



ภาพที่ 3.49 ขั้นตอนการสร้างหน้าต่างค้นหาข้อมูล ขั้นตอนที่ 6

g) ดับเบิลคลิกที่ SessionEJB_TestLocal.java ที่ปรากฏขึ้นมาใน System Navigator และพิมพ์โค้ดในส่วนของ Code Editor ดังภาพ



ภาพที่ 3.50 ขั้นตอนการสร้างหน้าต่างค้นหาข้อมูล ขั้นตอนที่ 7

โค้ดที่พิมพ์ในหน้า SessionEJB_TestLocal.java

```
package test.datamodel.ejb;

import java.util.List;

import javax.ejb.Local;

import test.datamodel.entities.EmployeesTemp;

@Local

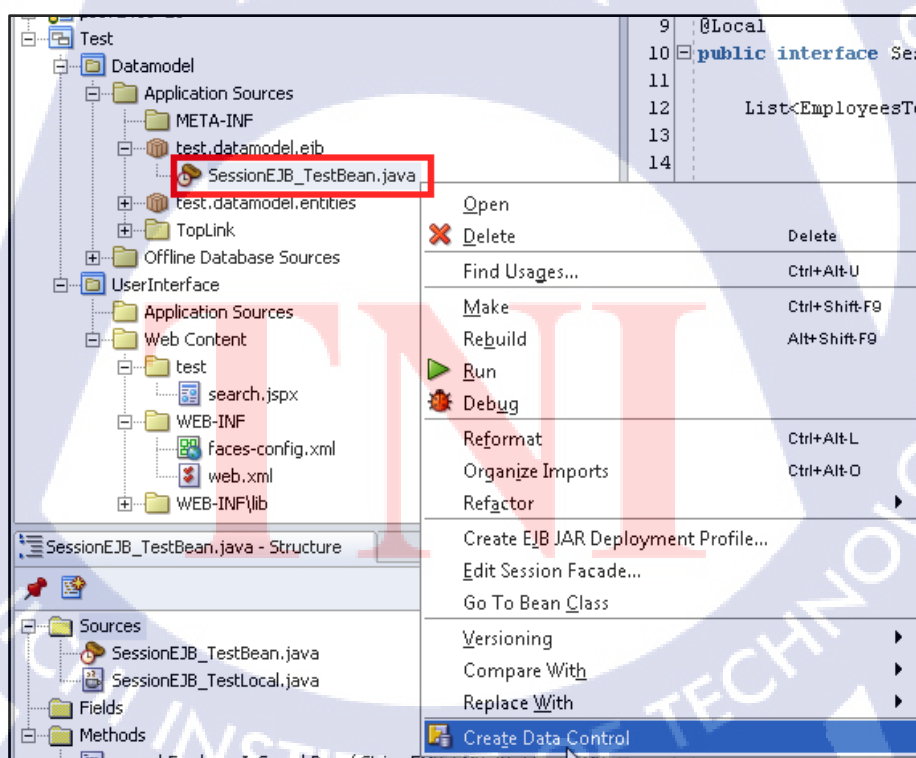
public interface SessionEJB_TestLocal {

    List<EmployeesTemp> searchEmployeeInSearchPage(String EMP_EMP_ID,

                                                    String EMP_TITLE_NAME,
                                                    String EMP_FIRST_NAME,
                                                    String EMP_LAST_NAME);

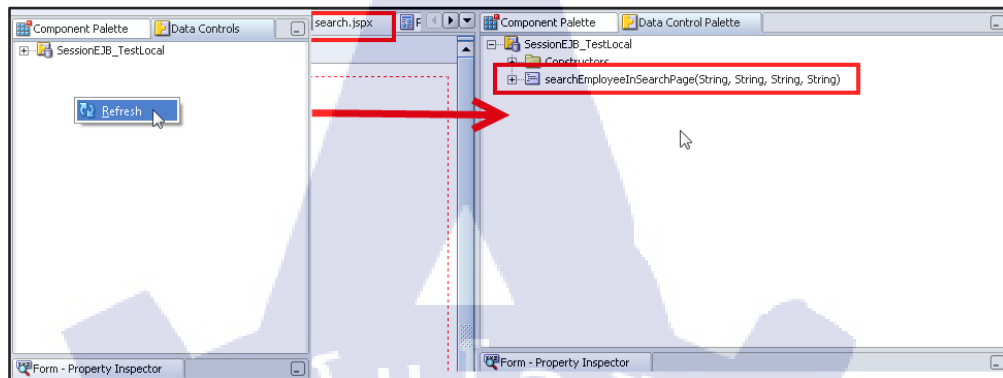
}
```

h) คลิกขวาที่ SessionEJB_TestBean.java เลือก Create Data Control ดังภาพ



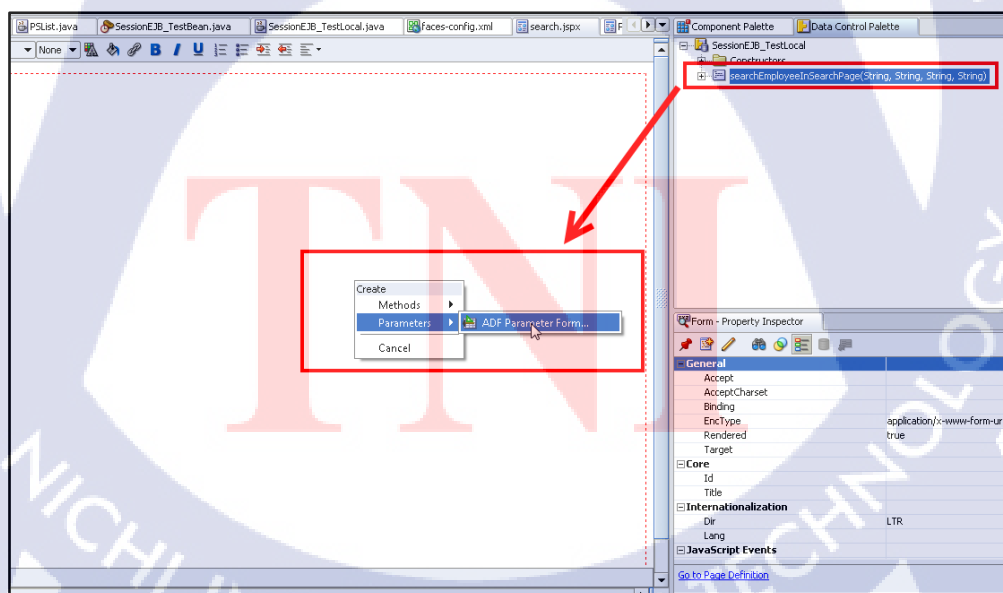
ภาพที่ 3.51 ขั้นตอนการสร้างหน้าค้นหาข้อมูล ขั้นตอนที่ 8

i) คลิกขวาที่ SessionEJB_TestLocal ใน Data Control Palette เลือก Refresh จะปรากฏดังภาพ



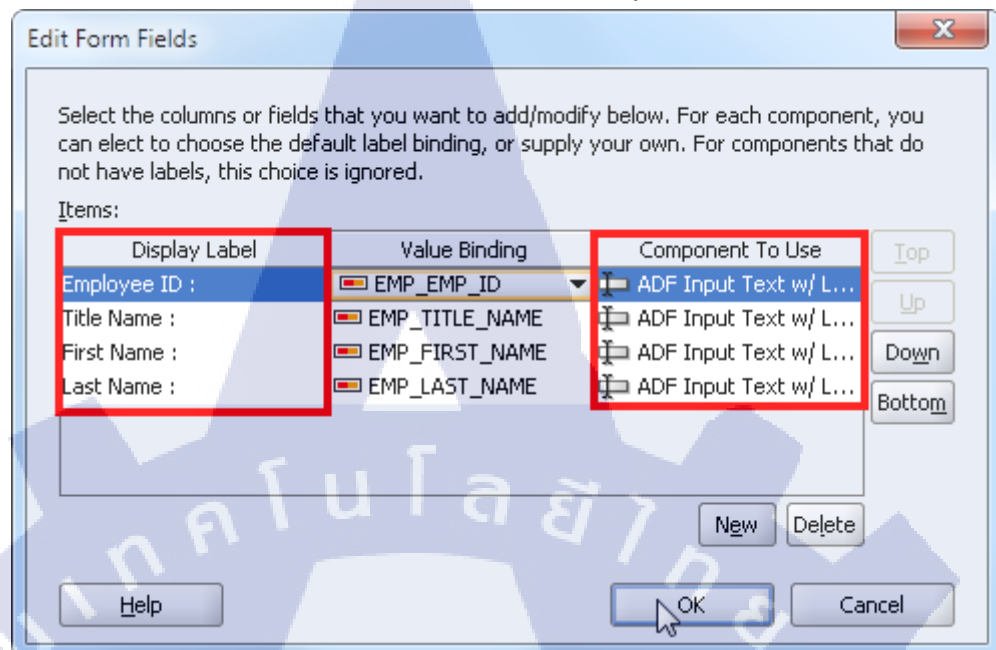
ภาพที่ 3.52 ขั้นตอนการสร้างหน้าค้นหาข้อมูล ขั้นตอนที่ 9

j) ลาก searchEmployeeInSearchPage ลงมาในแถบ Design ในส่วน Code Editor ของหน้า search.jspx จะขึ้นส่วนของ create ขึ้นมา เลือก Parameters → ADF Parameter Form... ดังภาพ

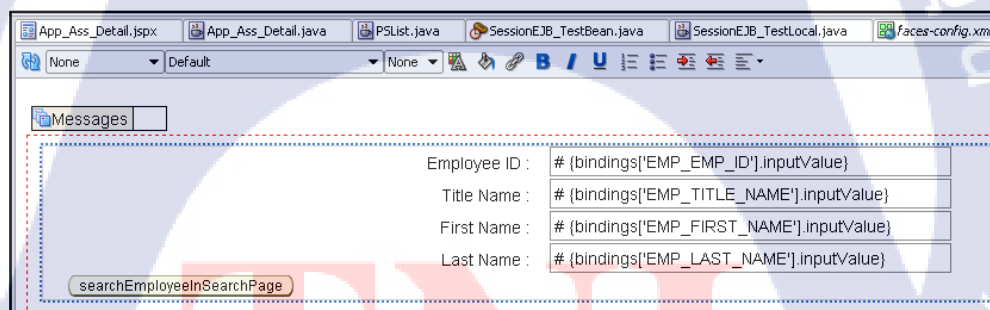


ภาพที่ 3.53 ขั้นตอนการสร้างหน้าค้นหาข้อมูล ขั้นตอนที่ 10

k) ตั้งชื่อ Label ดังภาพที่ 3.54 จะปรากฏดังภาพที่ 3.55

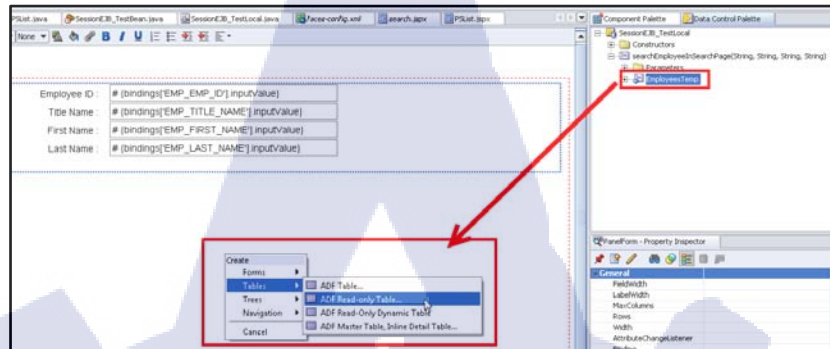


ภาพที่ 3.54 ขั้นตอนการสร้างหน้าค้นหาข้อมูล ขั้นตอนที่ 11



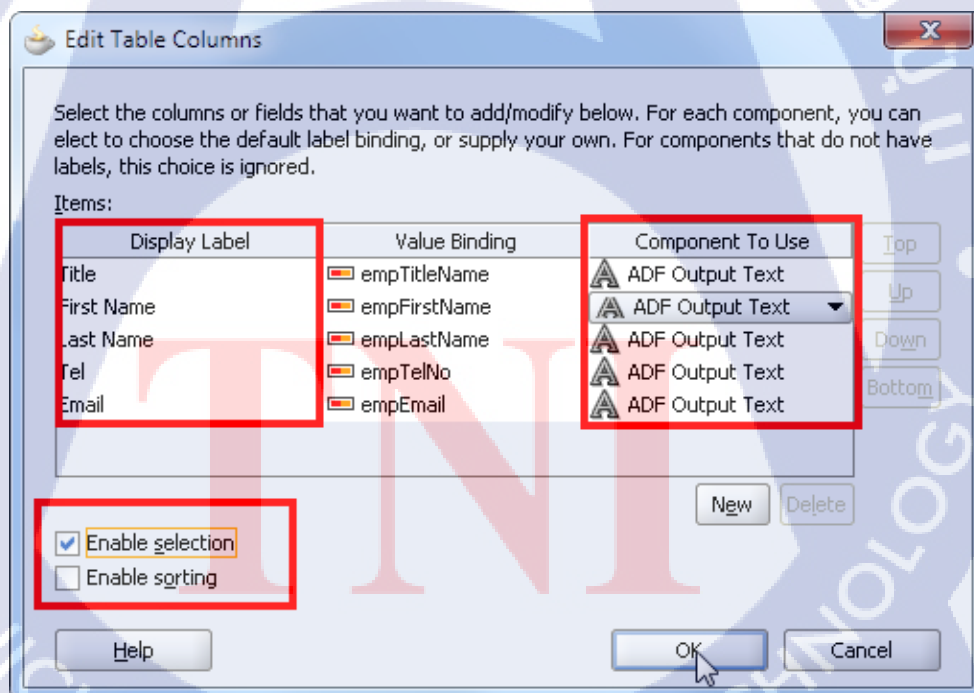
ภาพที่ 3.55 แสดงส่วน input Form ที่ถูกสร้างขึ้น

1) ลาก EmployeesTemp มาในส่วน Design ของ Code Editor จะปรากฏ
การ Create เลือก Table → ADF Read-only Tables.. ดังภาพ



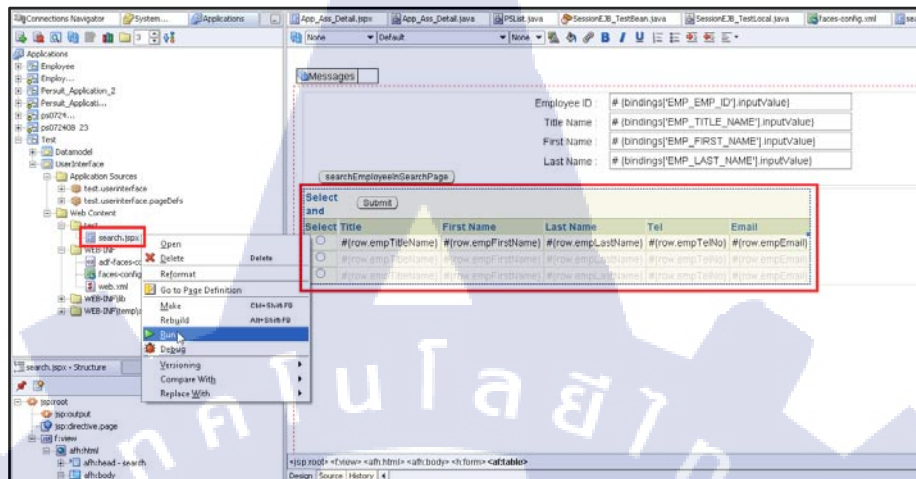
ภาพที่ 3.56 ขั้นตอนการสร้างหน้าค้นหาข้อมูล ขั้นตอนที่ 12

m) ตั้งชื่อ Label ดังภาพ



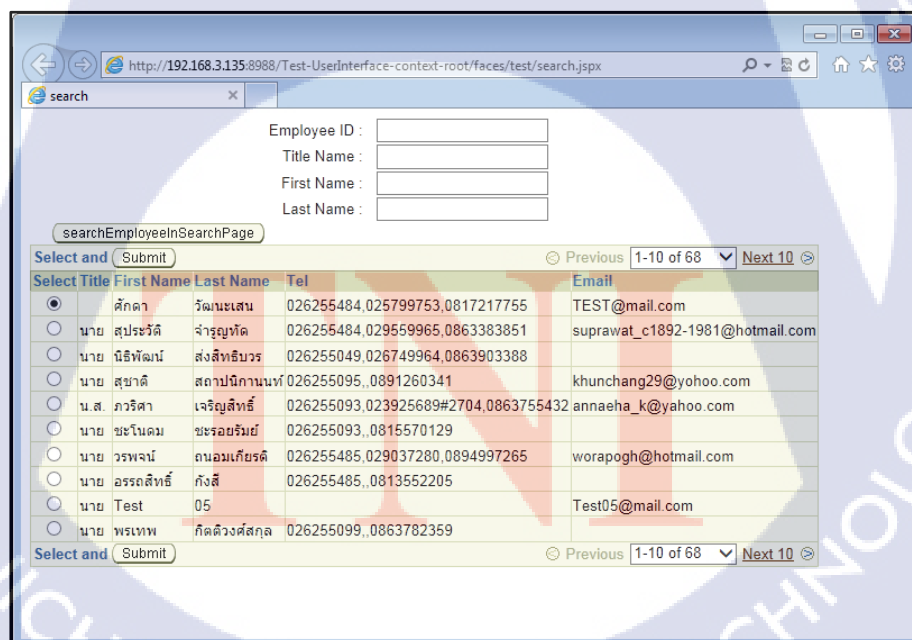
ภาพที่ 3.57 ขั้นตอนการสร้างหน้าค้นหาข้อมูล ขั้นตอนที่ 13

n) จะปรากฏตารางแสดงข้อมูลดังภาพ จากนั้นคลิกขวาที่ search.jspx ในแถบ Navigator จากนั้นเลือก Run ดังภาพ



ภาพที่ 3.58 ขั้นตอนการสร้างหน้าค้นหาข้อมูล ขั้นตอนที่ 14

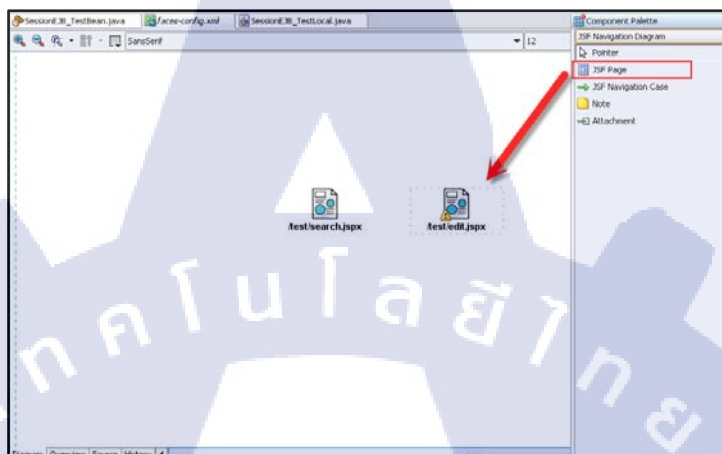
o)งานจะถูกรันขึ้นบน Web browser ดังภาพ



ภาพที่ 3.59 ผลการ Run ของหน้า search.jspx

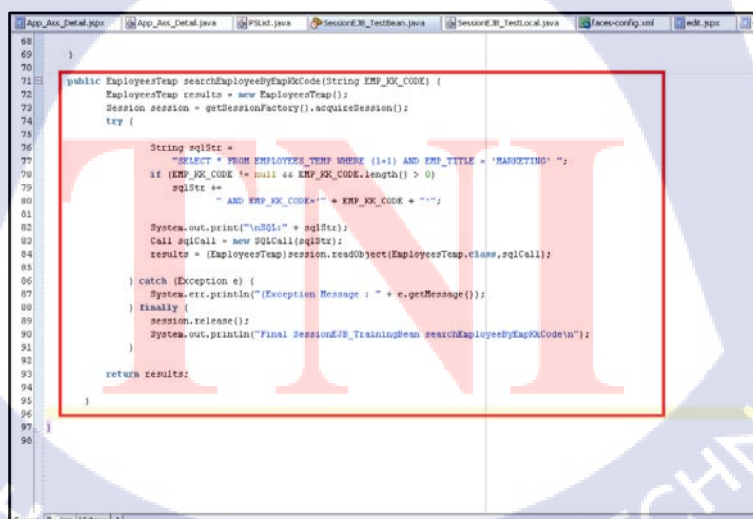
3.3.1.10 สร้างหน้าแก้ไขข้อมูล

a) ลาก JSF Page จาก Component Palette มายัง faces-config.xml ดังภาพ จากนั้นดับเบิลคลิกไอคอนและทำซ้ำขั้นตอนเช่นเดียวกันกับ serch.jspx เพื่อให้ได้ไฟล์ edit.jspx มา



ภาพที่ 3.60 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 1

b) พิมพ์โค้ดในหน้า SessionEJB_TestBean.java ต่อจากโค้ดเดิม ดังภาพ



ภาพที่ 3.61 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 2

โค้ดที่พิมพ์ในหน้า SessionEJB_TestBean.java

```
public EmployeesTemp searchEmployeeByEmpKkCode(String EMP_KK_CODE) {

    EmployeesTemp results = new EmployeesTemp();

    Session session = getSessionFactory().acquireSession();

    try {String sqlStr = "SELECT * FROM EMPLOYEES_TEMP WHERE (1=1) AND
EMP_TITLE = 'MARKETING' ";

        if (EMP_KK_CODE != null && EMP_KK_CODE.length() > 0)

            sqlStr += " AND EMP_KK_CODE='" + EMP_KK_CODE + "'";

        System.out.print("\nSQL:" + sqlStr);

        Call sqlCall = new SQLCall(sqlStr);

        results = (EmployeesTemp)session.readObject(EmployeesTemp.class,sqlCall);

    } catch (Exception e) {

        System.err.println("(Exception Message : " + e.getMessage());

    } finally {

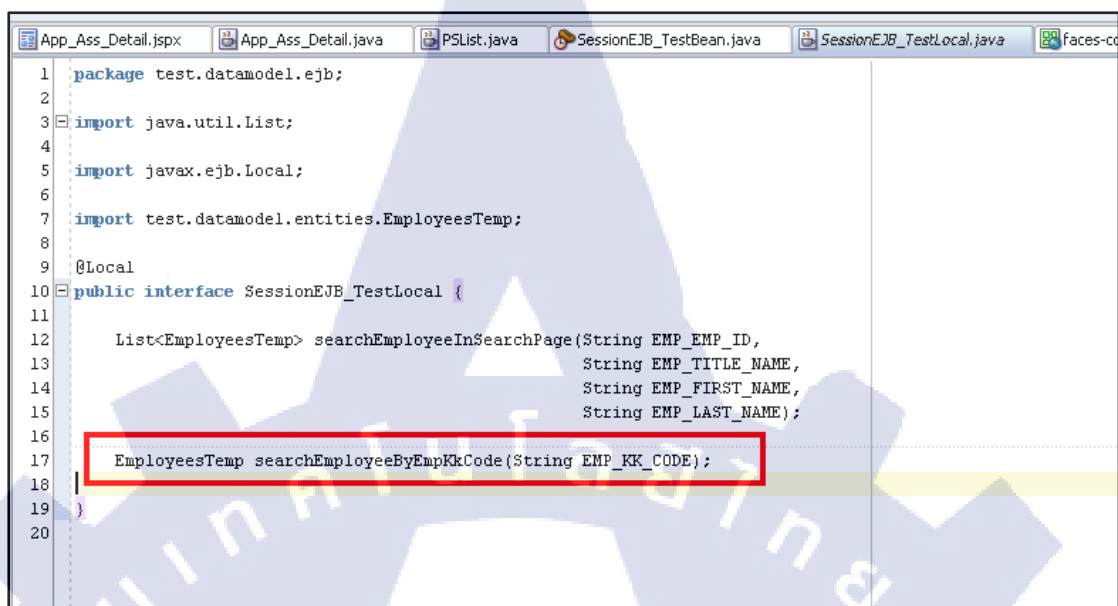
        session.release();

        System.out.println("Final SessionEJB_TrainingBean
searchEmployeeByEmpKkCode\n");

    }

    return results; }
```

c) พิมพ์โค้ดในหน้า SessionEJB_TestLocal.java ต่อจากโค้ดเดิม ดังภาพ



```

1 package test.datamodel.ejb;
2
3 import java.util.List;
4
5 import javax.ejb.Local;
6
7 import test.datamodel.entities.EmployeesTemp;
8
9 @Local
10 public interface SessionEJB_TestLocal {
11
12     List<EmployeesTemp> searchEmployeeInSearchPage(String EMP_EMP_ID,
13                                                    String EMP_TITLE_NAME,
14                                                    String EMP_FIRST_NAME,
15                                                    String EMP_LAST_NAME);
16
17     EmployeesTemp searchEmployeeByEmpKkCode(String EMP_KK_CODE);
18
19 }
20

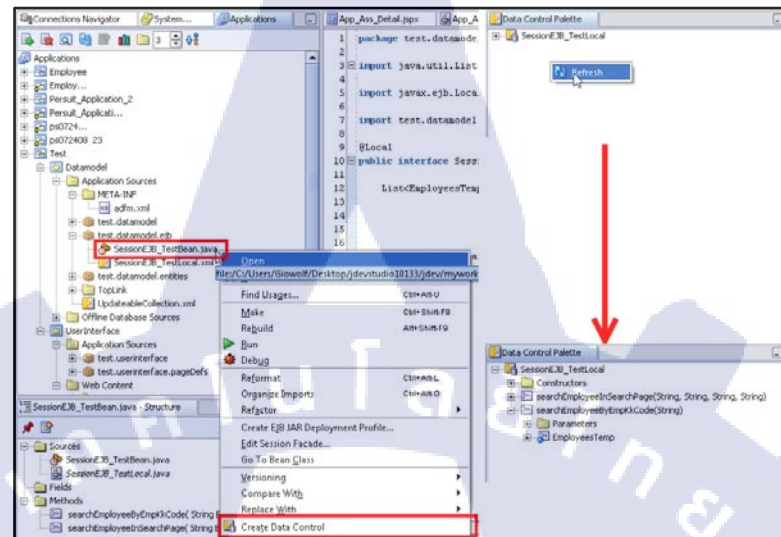
```

ภาพที่ 3.62 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 3

โค้ดที่พิมพ์ในหน้า SessionEJB_TestLocal.java

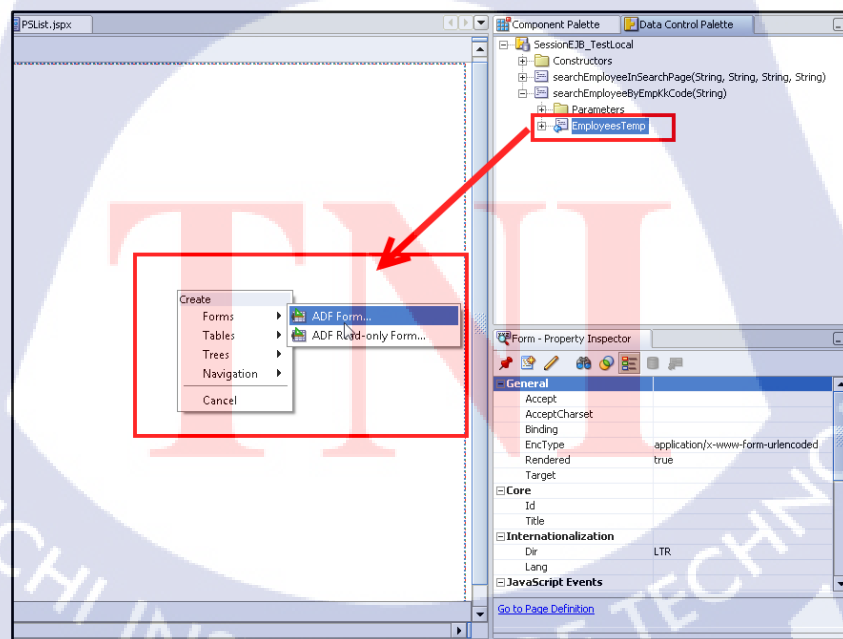
EmployeesTemp searchEmployeeByEmpKkCode(String EMP_KK_CODE);

d) คลิกขวาที่ SessionEJB_TestBean.java เลือก Create Data Control จากนั้นคลิกขวาแล้วเลือก Refresh ใน Data Control Palette จะปรากฏดังภาพ



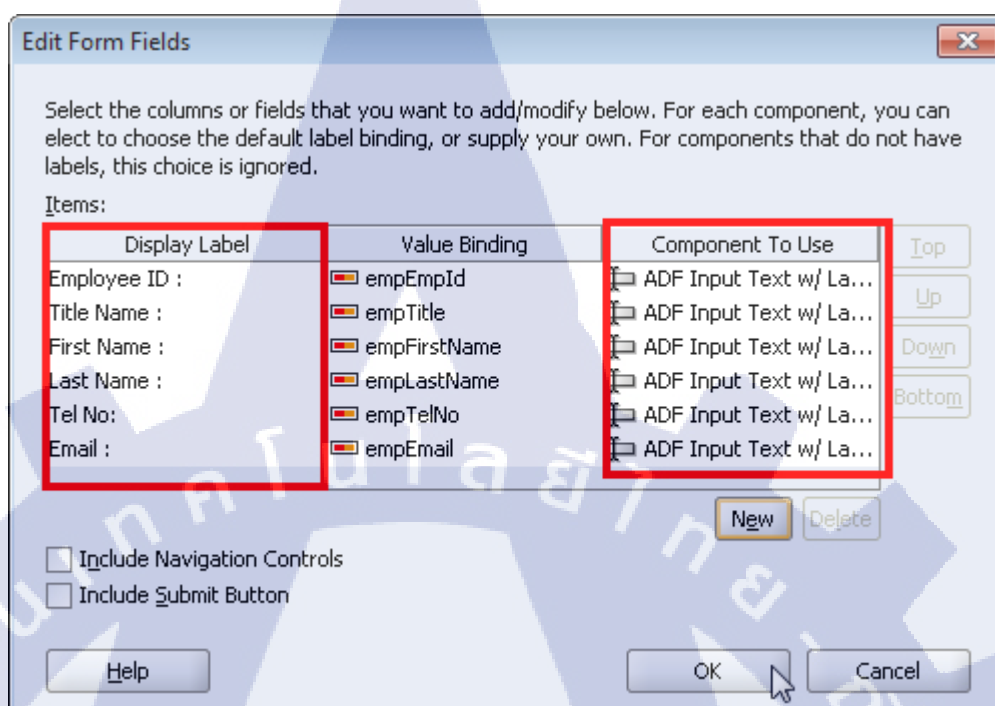
ภาพที่ 3.63 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 4

e) ลาก EmployeesTemp ลงมาในส่วน Design ของหน้า Edit.jspx จะปรากฏการ Create เลือก Forms → ADF Form.. ดังภาพ



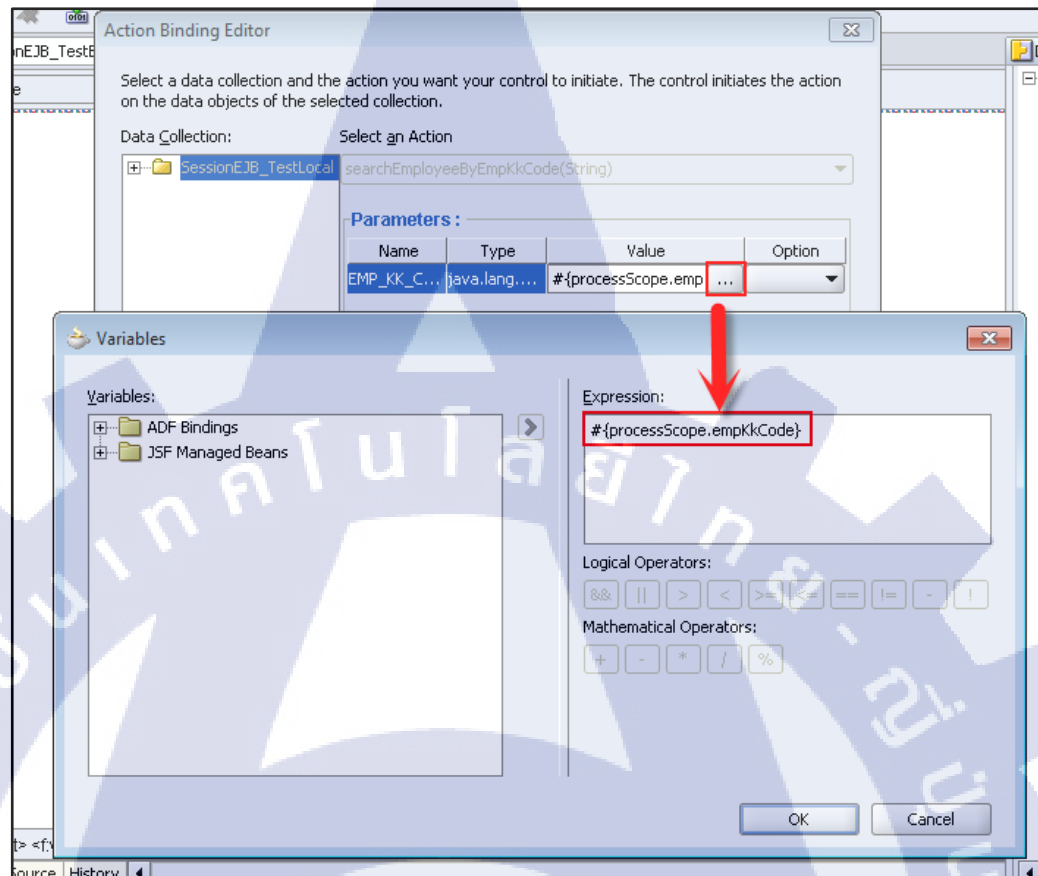
ภาพที่ 3.64 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 5

f) ตั้งชื่อ Label ดังภาพ



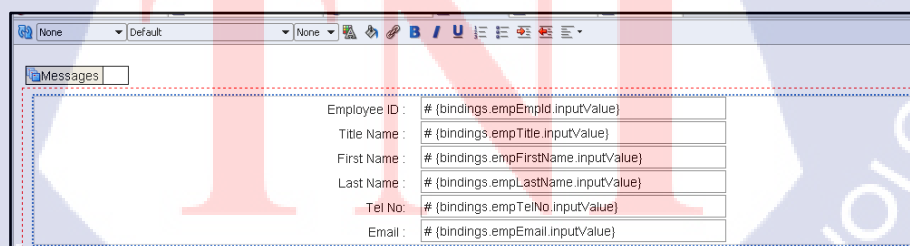
ภาพที่ 3.65 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 6

g) เลือก ... ในช่อง EMP_KK_CODE แล้วพิมพ์ดังภาพ



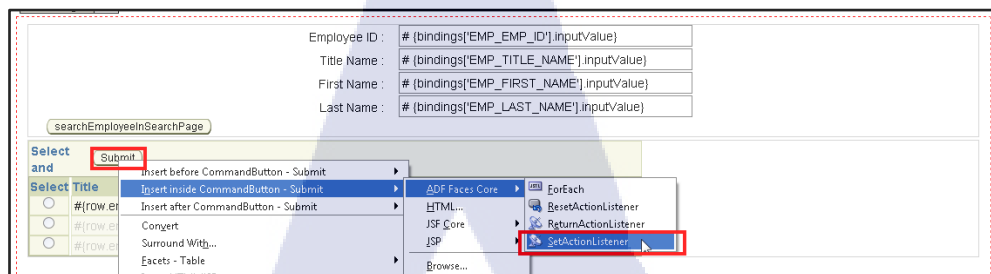
ภาพที่ 3.66 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 7

h) จะปรากฏในส่วนของ Design ของ Edit.jspx ดังภาพ



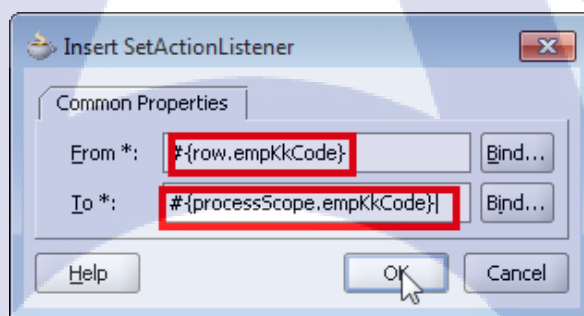
ภาพที่ 3.67 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 8

i) ไปยังหน้า Seach.jspxคลิกขวาที่ปุ่ม Submit เลือก Insert inside
 CommandButton-Submit → ADF Faces Core → SetActionListener



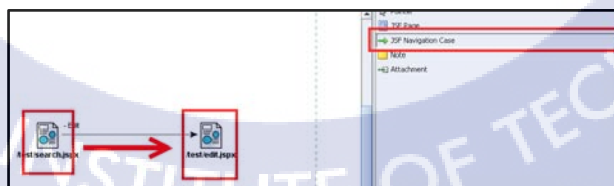
ภาพที่ 3.68 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 9

j) กรอกข้อมูลดังภาพ



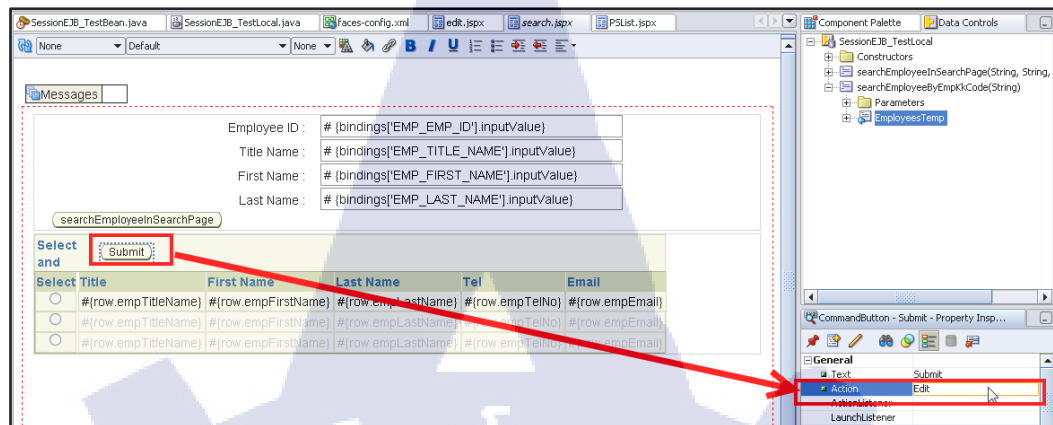
ภาพที่ 3.69 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 10

k) ลาก JSF Navigation Case (Component Palette) ในหน้า faces-config.xml เพื่อเชื่อมต่อหน้า และตั้งชื่อเส้นดังกล่าว



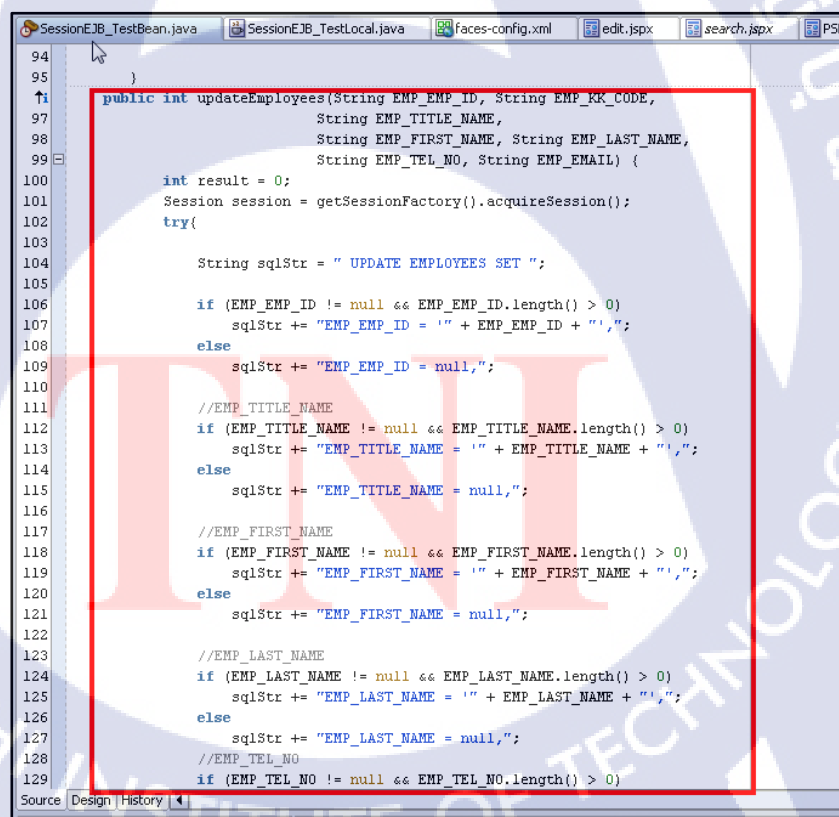
ภาพที่ 3.70 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 11

1) เลือก Action ให้ปุ่ม Submit เป็น Edit ในหน้าต่าง Property ดังภาพ



ภาพที่ 3.71 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 12

m) พิมพ์โค้ดในหน้า SessionEJB_TestBean.java ต่อจากโค้ดเดิม ดังภาพ



ภาพที่ 3.72 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 13

ได้ในหน้า SessionEJB_TestBean.java

```
public int updateEmployees(String EMP_EMP_ID, String EMP_KK_CODE,

    String EMP_TITLE_NAME,

    String EMP_FIRST_NAME, String EMP_LAST_NAME,

    String EMP_TEL_NO, String EMP_EMAIL) {

    int result = 0;

    Session session = getSessionFactory().acquireSession();

    try{

        String sqlStr = " UPDATE EMPLOYEES SET ";

        if (EMP_EMP_ID != null && EMP_EMP_ID.length() > 0)

            sqlStr += "EMP_EMP_ID = " + EMP_EMP_ID + ",";

        else

            sqlStr += "EMP_EMP_ID = null,";

        if (EMP_TITLE_NAME != null && EMP_TITLE_NAME.length() > 0)

            sqlStr += "EMP_TITLE_NAME = " + EMP_TITLE_NAME + ",";

        else

            sqlStr += "EMP_TITLE_NAME = null,";

        if (EMP_FIRST_NAME != null && EMP_FIRST_NAME.length() > 0)

            sqlStr += "EMP_FIRST_NAME = " + EMP_FIRST_NAME + ",";
```

```

else

    sqlStr += "EMP_FIRST_NAME = null,";

    if (EMP_LAST_NAME != null && EMP_LAST_NAME.length() > 0)

        sqlStr += "EMP_LAST_NAME = '" + EMP_LAST_NAME + "',";

    else

        sqlStr += "EMP_LAST_NAME = null,";

    if (EMP_TEL_NO != null && EMP_TEL_NO.length() > 0)

        sqlStr += "EMP_TEL_NO = '" + EMP_TEL_NO + "',";

    else

        sqlStr += "EMP_TEL_NO = null,";

    if (EMP_EMAIL != null && EMP_EMAIL.length() > 0)

        sqlStr += "EMP_EMAIL = '" + EMP_EMAIL + "'";

    else

        sqlStr += "EMP_EMAIL = null";

    sqlStr += " WHERE EMP_KK_CODE = '" + EMP_KK_CODE + "' ";

    System.out.print("\nSQL UPDATE EMPLOYEES:" + sqlStr);

    Call sqlCall = new SQLCall(sqlStr);

    result = session.executeNonSelectingCall(sqlCall);

} catch (Exception e) {

```

```

        System.out.println("\n Exception : " +e.getMessage() + "\n");

    } finally {

        System.out.println("Final SessionEJB_TrainingBean (updateEmployees) \n");

        session.release();

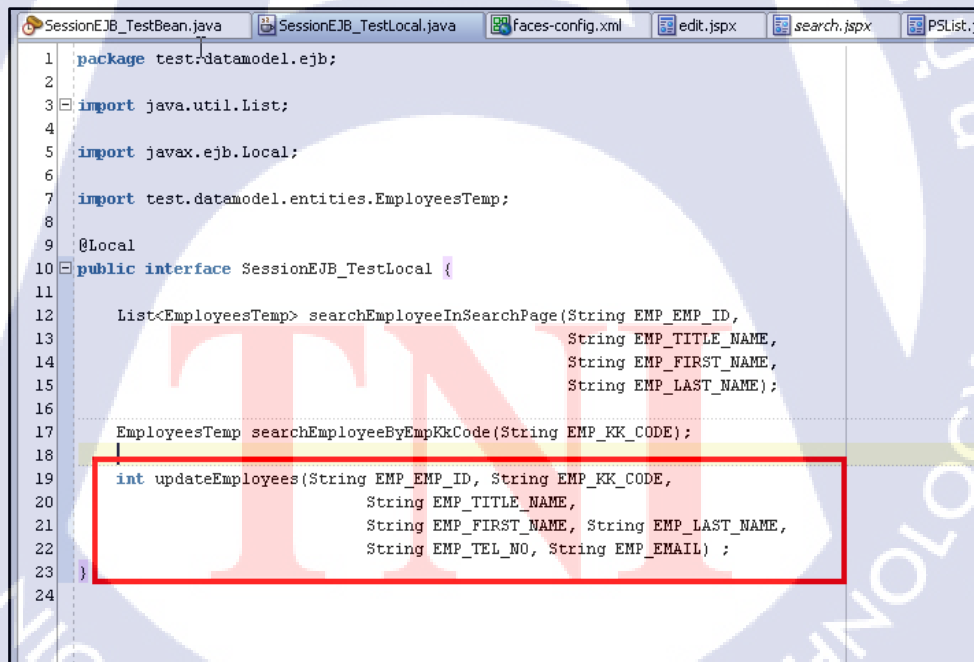
    }

    return result;

}

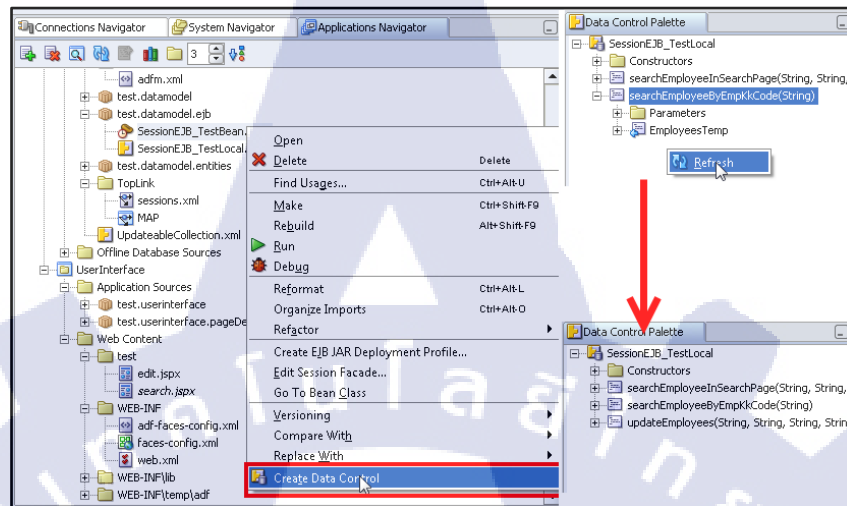
```

n) พิมพ์โค้ดในหน้า SessionEJB_TestLocal.java ต่อจากโค้ดเดิม ดังภาพ



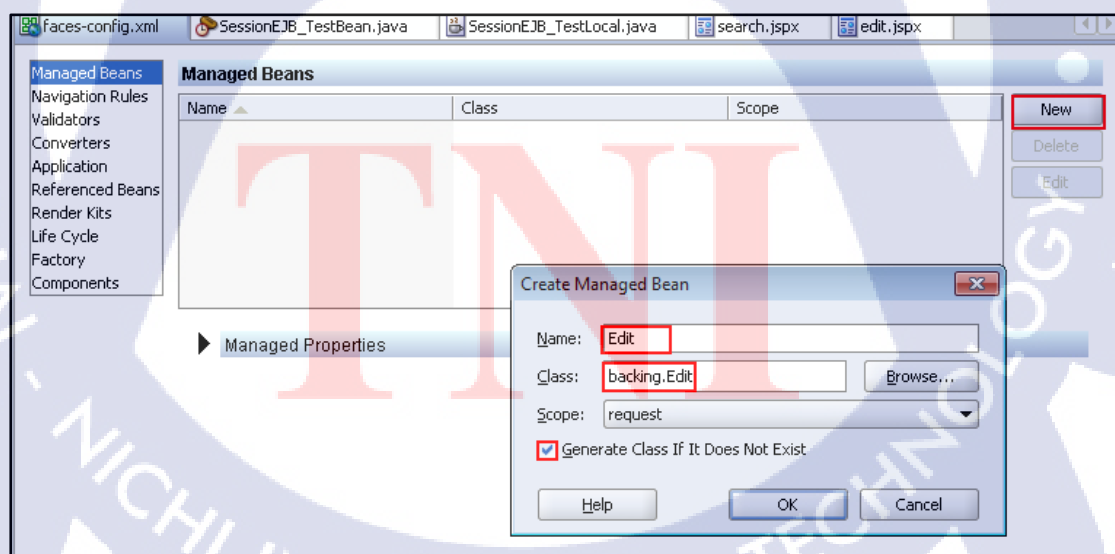
ภาพที่ 3.73 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 14

o)คลิกขวาที่ SessionEJB_TestBean.java เลือก Create Data Control จากนั้นคลิกขวาแล้วเลือก Refresh ใน Data Control Palette จะปรากฏดังภาพ



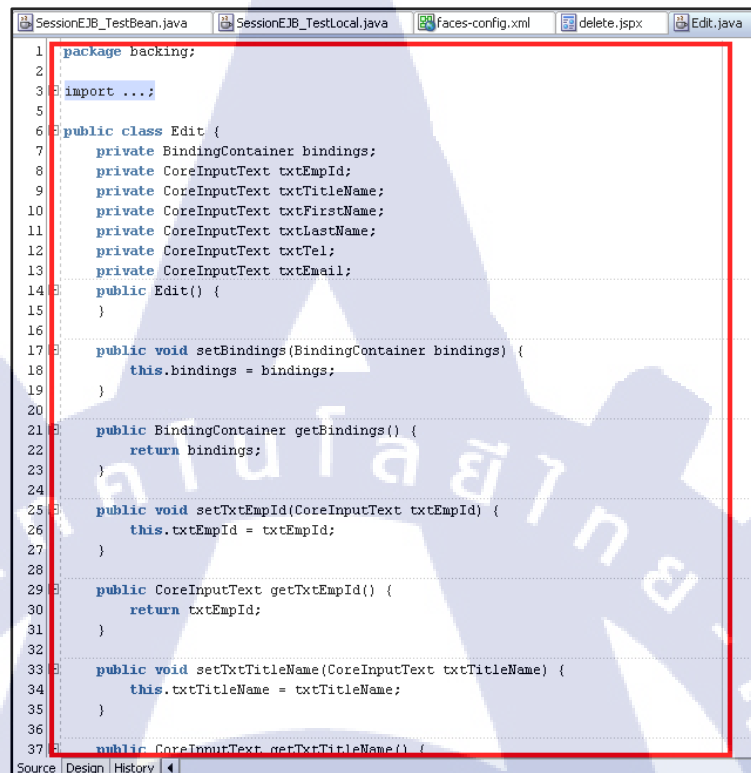
ภาพที่ 3.74 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 15

p)ไปที่ faces-config.xml เลือกแถบ overview ด้านล่างของส่วน Code Editor จากนั้นคลิกNewและกรอกข้อมูล แล้วกด OK ดังภาพ



ภาพที่ 3.75 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 16

q) พิมพ์โค้ดในหน้า Edit.java ดังภาพ



```

1 package backing;
2
3 import ...;
4
5
6 public class Edit {
7     private BindingContainer bindings;
8     private CoreInputText txtEmpId;
9     private CoreInputText txtTitleName;
10    private CoreInputText txtFirstName;
11    private CoreInputText txtLastName;
12    private CoreInputText txtTel;
13    private CoreInputText txtEmail;
14    public Edit() {
15    }
16
17    public void setBindings(BindingContainer bindings) {
18        this.bindings = bindings;
19    }
20
21    public BindingContainer getBindings() {
22        return bindings;
23    }
24
25    public void setTxtEmpId(CoreInputText txtEmpId) {
26        this.txtEmpId = txtEmpId;
27    }
28
29    public CoreInputText getTxtEmpId() {
30        return txtEmpId;
31    }
32
33    public void setTxtTitleName(CoreInputText txtTitleName) {
34        this.txtTitleName = txtTitleName;
35    }
36
37    public CoreInputText getTxtTitleName() {

```

ภาพที่ 3.76 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 17

โค้ดในหน้า Edit.java

```

package backing;

import oracle.adf.model.BindingContainer;
import oracle.adf.view.faces.component.core.input.CoreInputText;

public class Edit {
    private BindingContainer bindings;
    private CoreInputText txtEmpId;
    private CoreInputText txtTitleName;
    private CoreInputText txtFirstName;
    private CoreInputText txtLastName;

```

```
private CoreInputText txtTel;
private CoreInputText txtEmail;
public Edit() { }

public void setBindings(BindingContainer bindings) {
    this.bindings = bindings;
}

public BindingContainer getBindings() {
    return bindings;
}

public void setTxtEmpId(CoreInputText txtEmpId) {
    this.txtEmpId = txtEmpId;
}

public CoreInputText getTxtEmpId() {
    return txtEmpId;
}

public void setTxtTitleName(CoreInputText txtTitleName) {
    this.txtTitleName = txtTitleName;
}

public CoreInputText getTxtTitleName() {
    return txtTitleName;
}

public void setTxtFirstName(CoreInputText txtFirstName) {
    this.txtFirstName = txtFirstName;
}
```



```
}

public CoreInputText getTxtFirstName() {
    return txtFirstName;
}

public void setTxtLastName(CoreInputText txtLastName) {
    this.txtLastName = txtLastName;
}

public CoreInputText getTxtLastName() {
    return txtLastName;
}

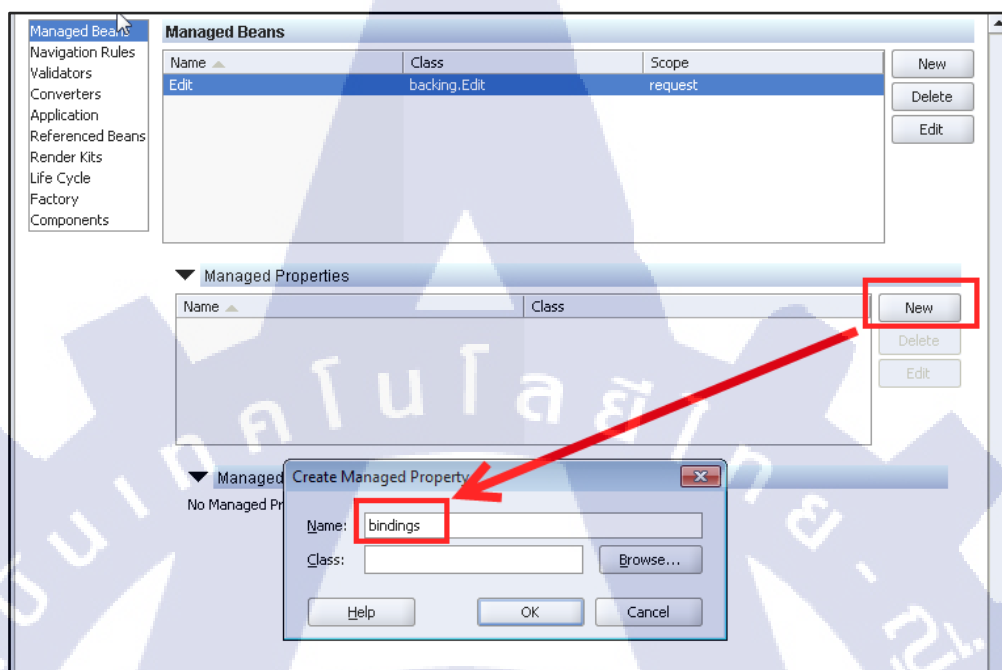
public void setTxtTel(CoreInputText txtTel) {
    this.txtTel = txtTel;
}

public CoreInputText getTxtTel() {
    return txtTel;
}

public void setTxtEmail(CoreInputText txtEmail) {
    this.txtEmail = txtEmail;
}

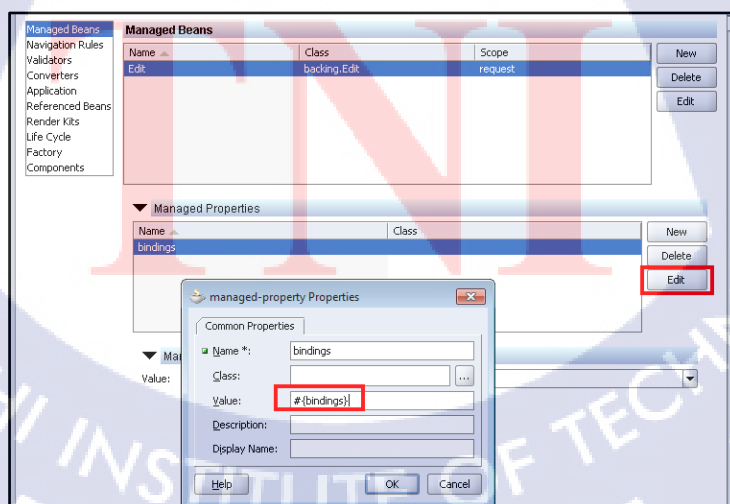
public CoreInputText getTxtEmail() {
    return txtEmail;
}
}
```

q) ส่วนของ overview ของ faces-config.xml คลิก New ในส่วนของ Managed Properties แล้วกรอก bindings ใน Name: จากนั้นกด OK ดังภาพ



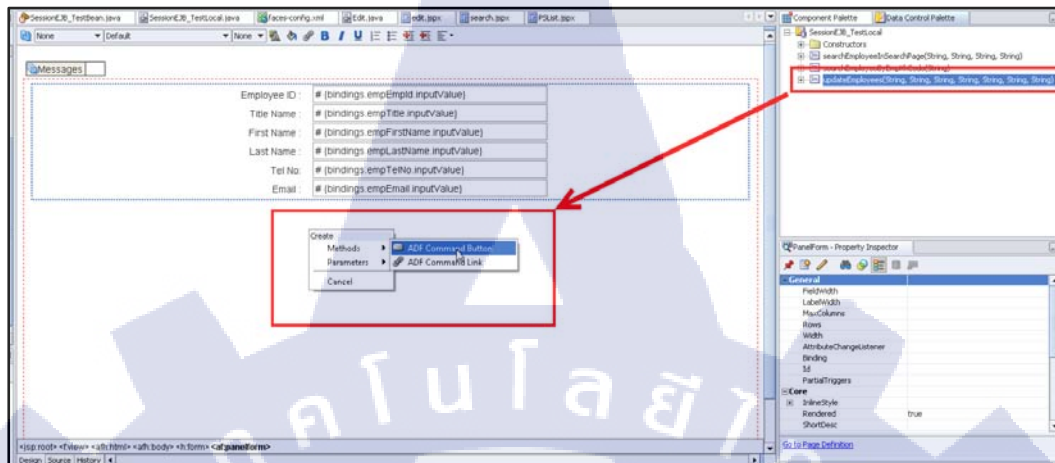
ภาพที่ 3.77 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 18

r) เลือก bindings คลิก edit กรอก Value: `#{bindings}` จากนั้นคลิก OK ดังภาพ



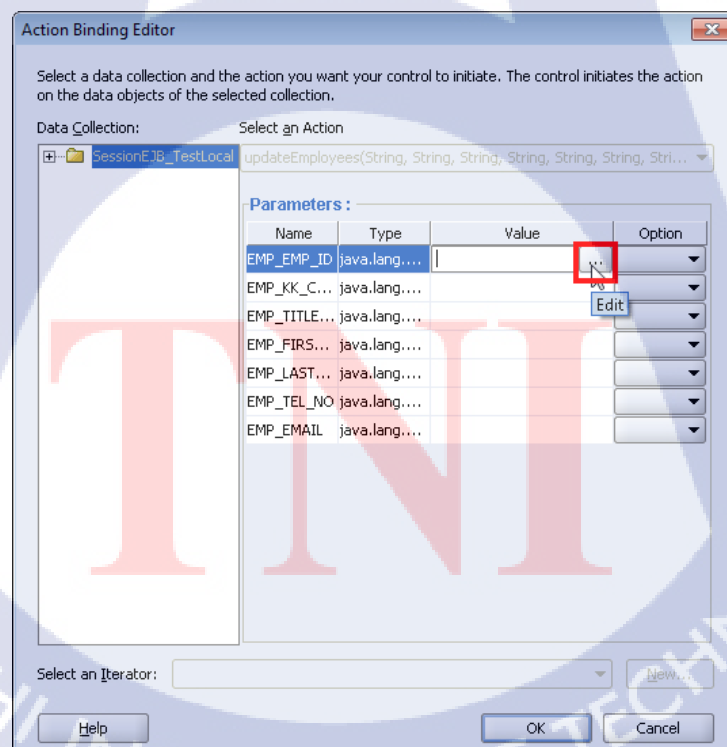
ภาพที่ 3.78 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 19

u) ลากupdateEmployeesจาก Data Control Palette จากนั้น create Parameters → ADF Command Button ดังภาพ



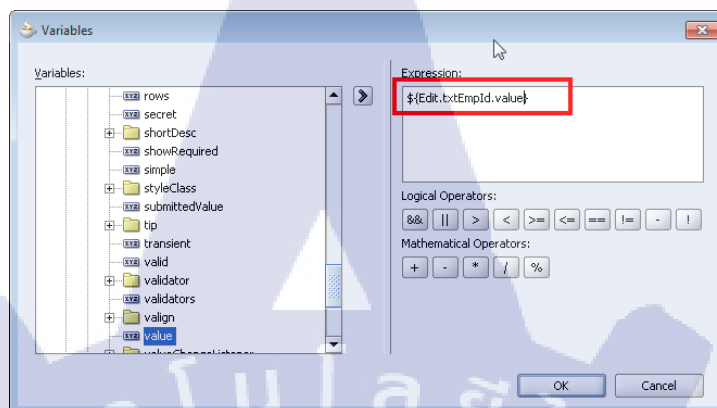
ภาพที่ 3.79 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 20

v) เลือกปุ่ม ... ดังภาพ



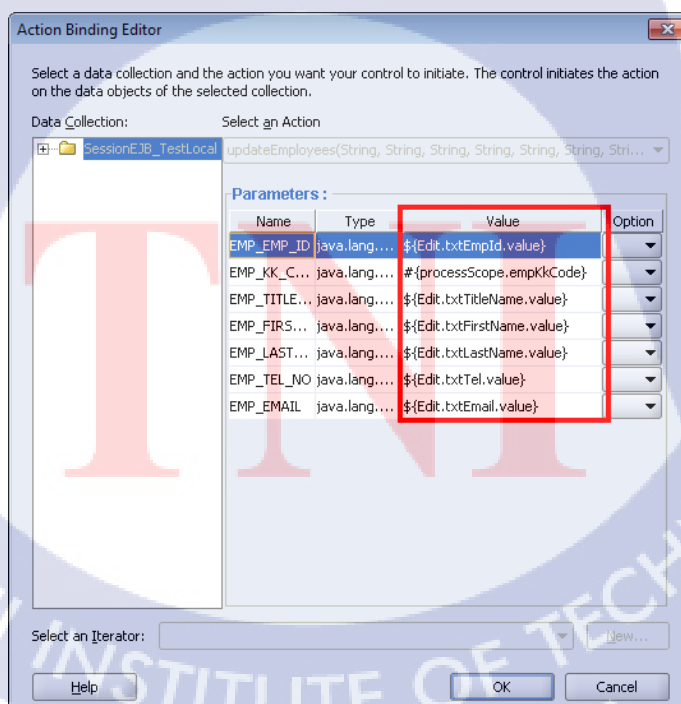
ภาพที่ 3.80 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 21

w) JDF Managed Beans → txtEmpId → value แล้วดับเบิ้ลคลิก จะขึ้น
โค้ดในช่อง Expression: ให้อัตโนมัติ จากนั้นคลิก OK ดังภาพ



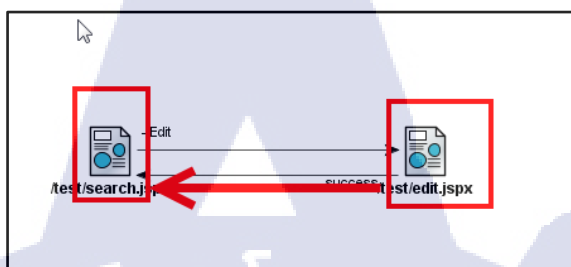
ภาพที่ 3.81 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 22

x) ทำเช่นเดียวกันกับ Parameters ที่เหลือ ยกเว้น EMP_KK_CODE ให้พิมพ์ Value เอง # {processScope.empKkCode} จากนั้นคลิก OK ดังภาพ



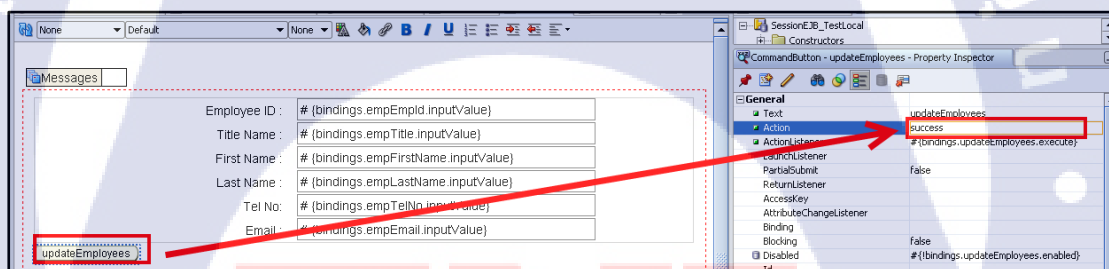
ภาพที่ 3.82 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 23

y) ลาก JSF Navigation Case (Component Palette) ในหน้า faces-config.xml เพื่อเชื่อมต่อหน้า และตั้งชื่อเส้นดังกล่าว



ภาพที่ 3.83 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 24

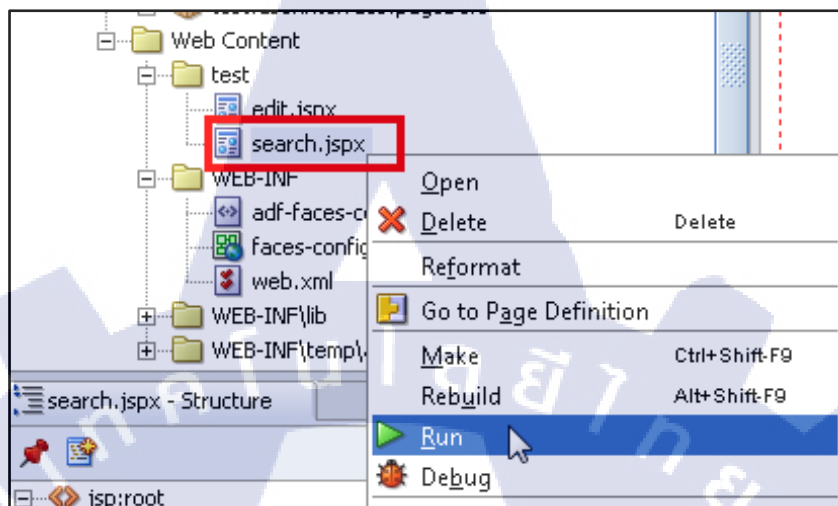
z) กำหนด Action ในส่วน property ของปุ่ม updateEmployees ในหน้า edit.jspx เป็น success ดังภาพ



ภาพที่ 3.84 ขั้นตอนการสร้างหน้าแก้ไขข้อมูล ขั้นตอนที่ 25

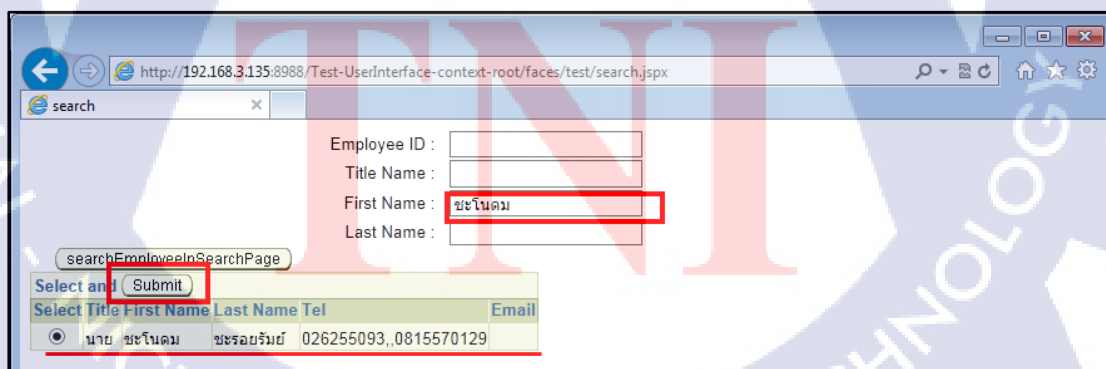
Run and Result

- 1) คลิกขวาที่ search.jspx ใน Navigator เลือก Run ดังภาพ



ภาพที่ 3.85 แสดงวิธีการ Run

- 2) ลองกรอก **ชื่อนาม** ในช่อง First Name: แล้วคลิกปุ่ม SearchEmployeeInSearchPage ตารางจะแสดงเพียงแค่ข้อมูลที่มีชื่อตามที่กรอกไว้ว่ามีกี่ข้อมูล จากตัวอย่างมีเพียงข้อมูลเดียว จากนั้นลองเลือกข้อมูลแล้วกด Submit ดังภาพที่ 3.68



ภาพที่ 3.86 แสดงการทดลองใช้งานของหน้าค้นหาข้อมูล 1

Employee ID :

Title Name :

First Name :

Last Name :

searchEmployeeInSearchPage

Select and

Select	Title	First Name	Last Name	Tel	Email
☒	นาย	ชะโนดม	ชะรอยรัมย์	026255093,0815570129	TEST

ภาพที่ 3.87 แสดงการทดลองใช้งานของหน้าค้นหาข้อมูล 2

3) ลองแก้ไขข้อมูลในช่องใดก็ได้ แล้วคลิกปุ่ม updateEmployees ดังภาพที่ 3.88 จะปรากฏผลลัพธ์ ดังภาพที่ 3.89

Employee ID :

Title Name :

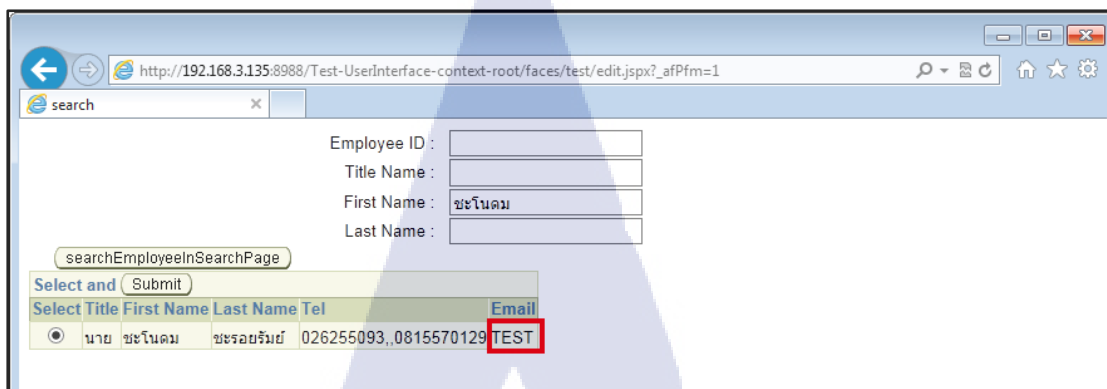
First Name :

Last Name :

Tel No:

Email :

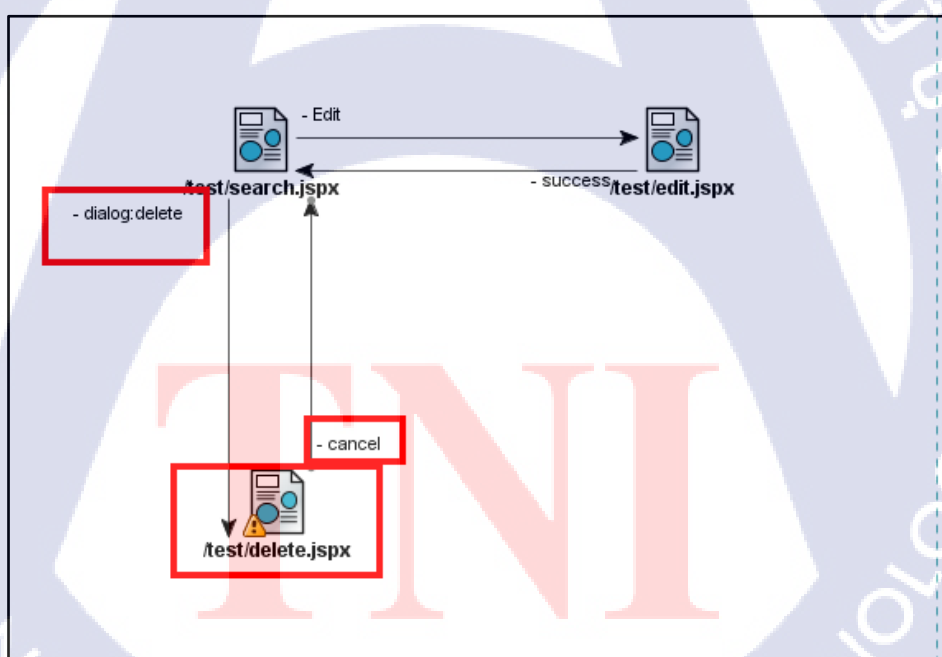
ภาพที่ 3.88 แสดงการทดลองใช้งานของหน้าค้นหาข้อมูล 3



ภาพที่ 3.89 แสดงการทดลองใช้งานของหน้าค้นหาข้อมูล 4

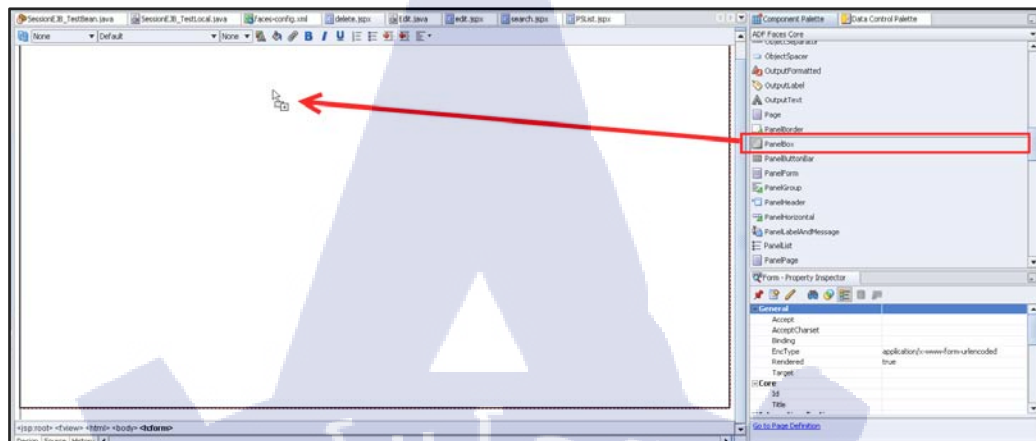
3.3.1.11 สร้างหน้าลบข้อมูล

1) สร้างหน้า delete.jspx และโยงเส้นเชื่อมหน้าดังกล่าว



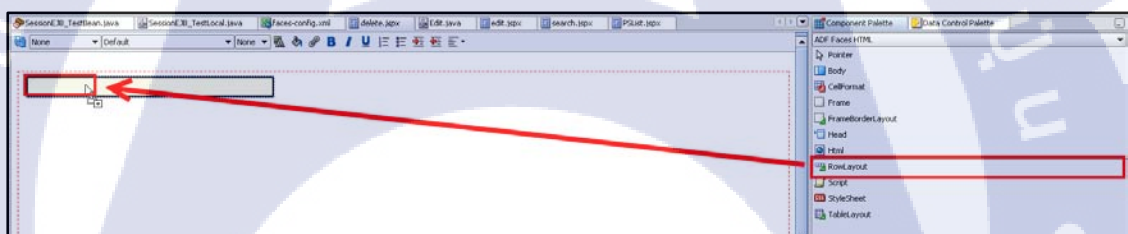
ภาพที่ 3.90 แสดงการสร้างหน้าลบข้อมูล ขั้นตอนที่ 1

2) ลาก PanelBox มาใส่ในหน้า delete.jpsx



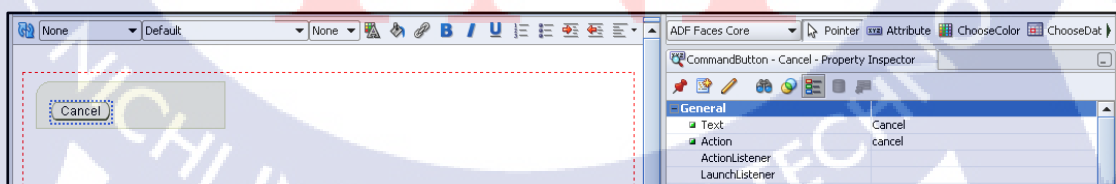
ภาพที่ 3.91 แสดงการสร้างหน้าลบข้อมูล ขั้นตอนที่ 2

3) ลาก RowLayout มาใส่ใน PanelBox



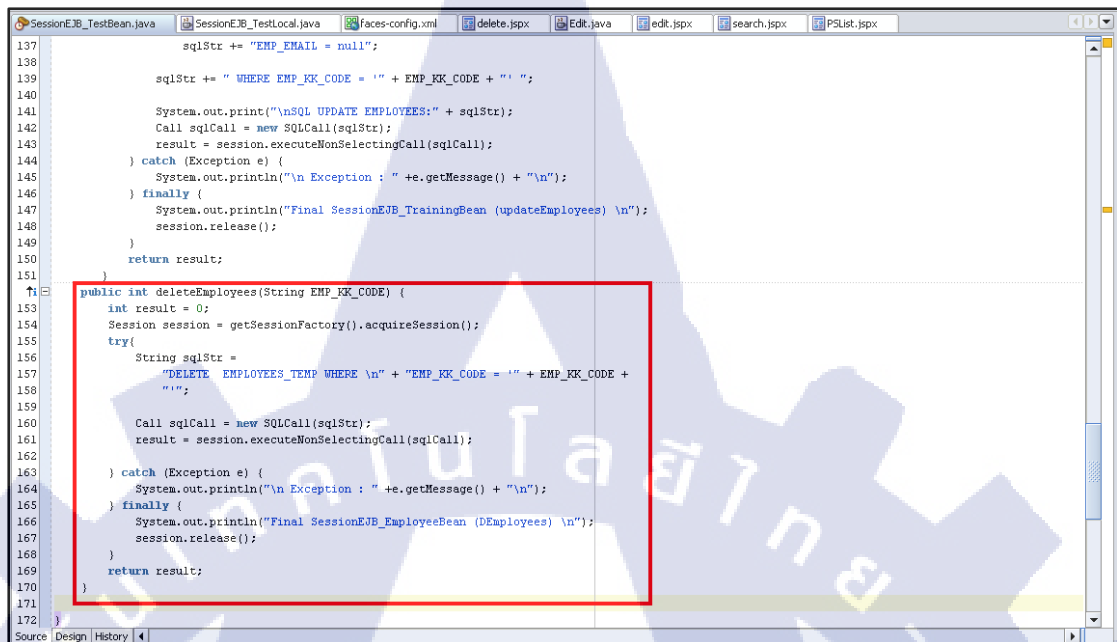
ภาพที่ 3.92 แสดงการสร้างหน้าลบข้อมูล ขั้นตอนที่ 3

4) ลาก CommandButton มาใส่ใน RowLayout และตั้งชื่อว่า Cancel ตั้งค่า Action: cancel



ภาพที่ 3.93 แสดงการสร้างหน้าลบข้อมูล ขั้นตอนที่ 4

5) พิมพ์โค้ดในหน้า SessionEJB_TestBean.java ต่อจากโค้ดเดิม

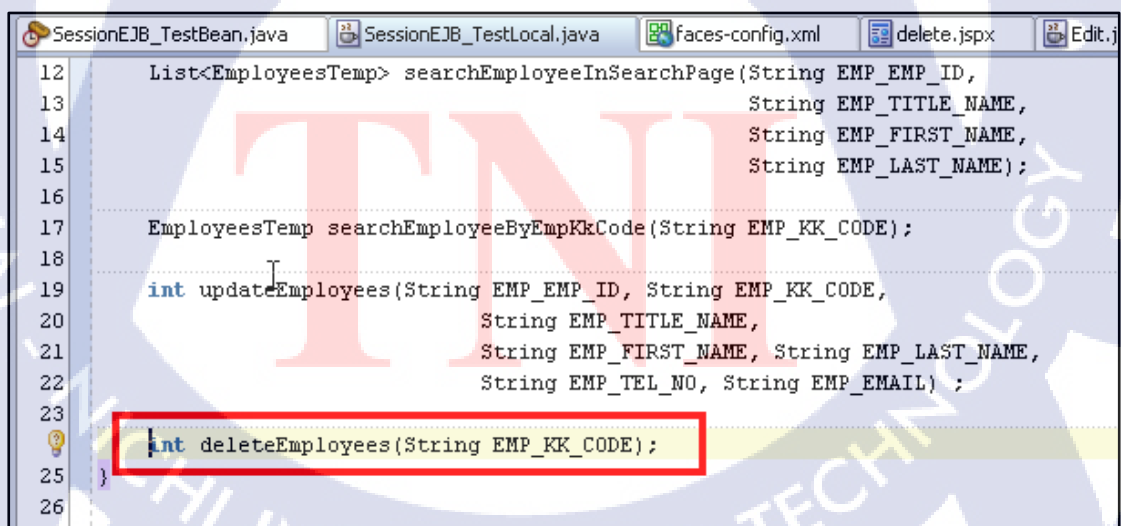


```

137      sqlStr += "EMP_EMAIL = null";
138
139      sqlStr += " WHERE EMP_KK_CODE = '" + EMP_KK_CODE + "' ";
140
141      System.out.println("SQL UPDATE EMPLOYEES:" + sqlStr);
142      Call sqlCall = new SQLCall(sqlStr);
143      result = session.executeNonSelectingCall(sqlCall);
144      catch (Exception e) {
145          System.out.println("\n Exception : " + e.getMessage() + "\n");
146      } finally {
147          System.out.println("Final SessionEJB_TrainingBean (updateEmployees) \n");
148          session.release();
149      }
150      return result;
151  }
152
153  public int deleteEmployees(String EMP_KK_CODE) {
154      int result = 0;
155      Session session = getSessionFactory().acquireSession();
156      try {
157          String sqlStr =
158              "DELETE EMPLOYEES_TEMP WHERE \n" + "EMP_KK_CODE = '" + EMP_KK_CODE +
159              "' ";
160
161          Call sqlCall = new SQLCall(sqlStr);
162          result = session.executeNonSelectingCall(sqlCall);
163      } catch (Exception e) {
164          System.out.println("\n Exception : " + e.getMessage() + "\n");
165      } finally {
166          System.out.println("Final SessionEJB_EmployeeBean (DEmployees) \n");
167          session.release();
168      }
169      return result;
170  }
171  }
172  }
  
```

ภาพที่ 3.94 แสดงการสร้างหน้าลบข้อมูล ขั้นตอนที่ 5

6) พิมพ์โค้ดในหน้า SessionEJB_TestLocal.java ต่อจากโค้ดเดิม



```

12      List<EmployeesTemp> searchEmployeeInSearchPage(String EMP_EMP_ID,
13                                                         String EMP_TITLE_NAME,
14                                                         String EMP_FIRST_NAME,
15                                                         String EMP_LAST_NAME);
16
17      EmployeesTemp searchEmployeeByEmpKkCode(String EMP_KK_CODE);
18
19      int updateEmployees(String EMP_EMP_ID, String EMP_KK_CODE,
20                          String EMP_TITLE_NAME,
21                          String EMP_FIRST_NAME, String EMP_LAST_NAME,
22                          String EMP_TEL_NO, String EMP_EMAIL) ;
23
24      int deleteEmployees(String EMP_KK_CODE);
25  }
26  }
  
```

ภาพที่ 3.95 แสดงการสร้างหน้าลบข้อมูล ขั้นตอนที่ 6

โค้ดที่พิมพ์ในหน้า SessionEJB_TestBean.java

```
public int deleteEmployees(String EMP_KK_CODE) {

    int result = 0;

    Session session = getSessionFactory().acquireSession();

    try{

        String sqlStr = "DELETE EMPLOYEES_TEMP WHERE \n" + "EMP_KK_CODE = '" +
EMP_KK_CODE + "'";

        Call sqlCall = new SQLCall(sqlStr);

        result = session.executeNonSelectingCall(sqlCall);

    } catch (Exception e) {

        System.out.println("\n Exception : " +e.getMessage() + "\n");

    } finally {

        System.out.println("Final SessionEJB_EmployeeBean (DEmployees) \n");

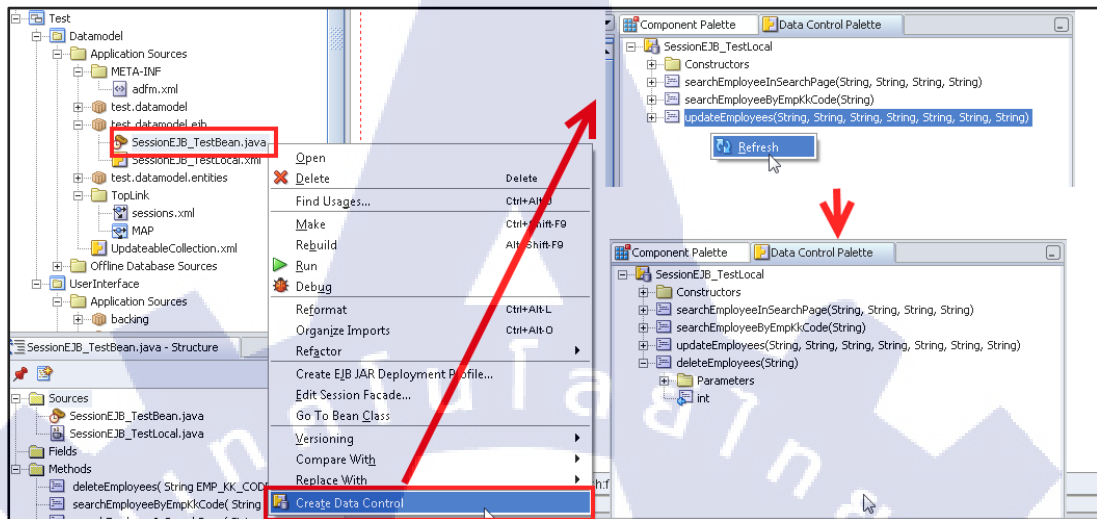
        session.release();

    }

    return result;

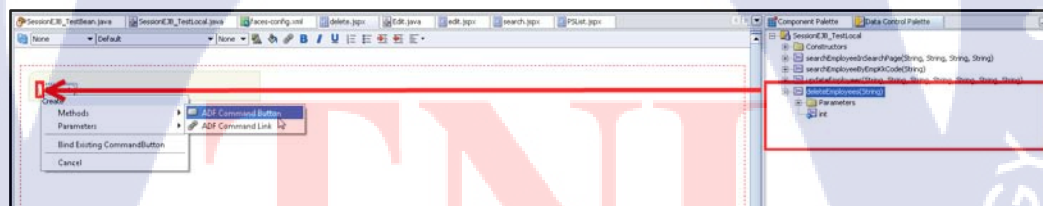
}
```

7) คลิกขวาที่ SessionEJB_TestBean.java เลือก Create Data Control จากนั้นคลิกขวาแล้วเลือก Refresh ใน Data Control Palette จะปรากฏ deleteEmployees(String)



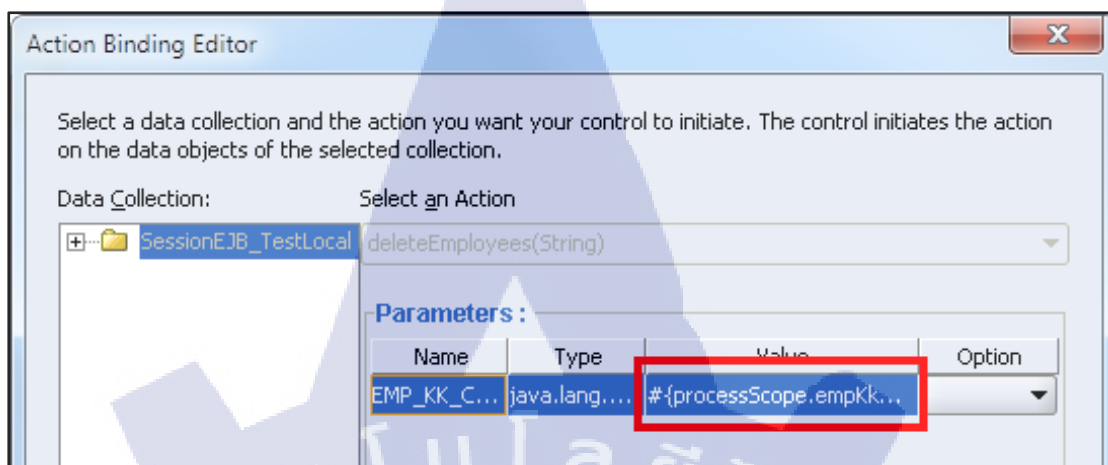
ภาพ3.96 แสดงการสร้างหน้าลบข้อมูล ขั้นตอนที่ 7

8) ลาก deleteEmployees ไปไว้หน้าปุ่ม Cancel แล้ว Create → Methods → ADF Command Button



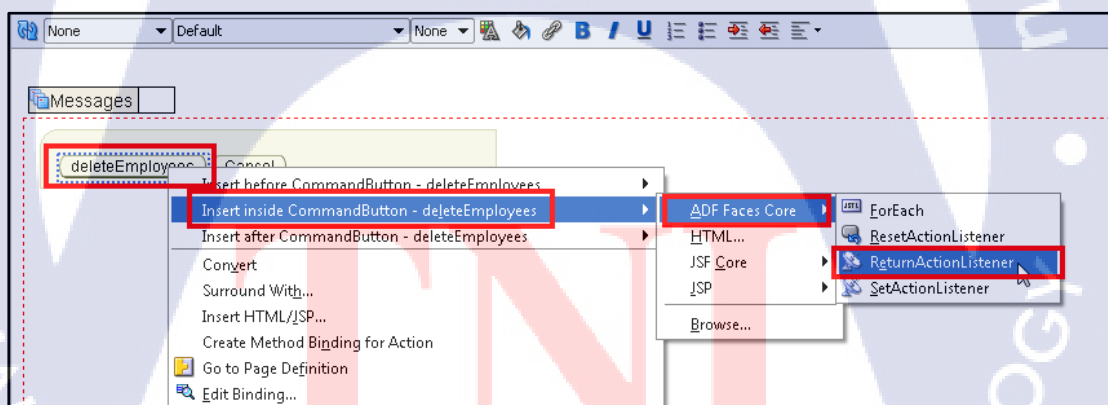
ภาพที่ 3.97 แสดงการสร้างหน้าลบข้อมูล ขั้นตอนที่ 8

9) กรอก #{processScope.empKkCode} ในช่อง Value



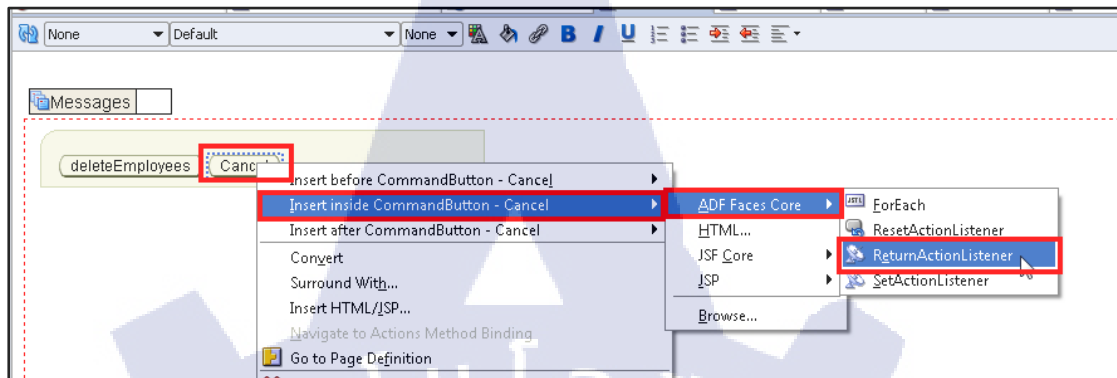
ภาพที่ 3.98 แสดงการสร้างหน้าลบข้อมูล ขั้นตอนที่ 9

10) คลิกขวาที่ปุ่ม deleteEmployees เลือก Insert inside CommandButton - deleteEmployees
 deleteEmployees → ADF Faces Core → ReturnActionListener



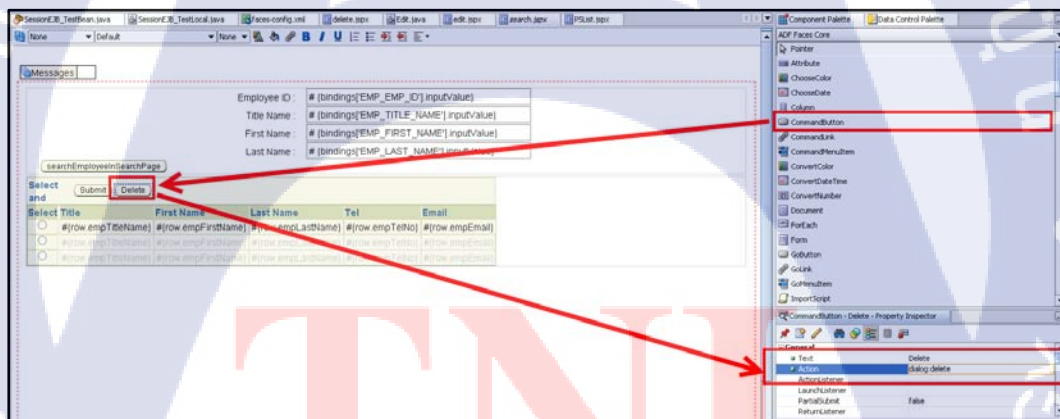
ภาพที่ 3.99 แสดงการสร้างหน้าลบข้อมูล ขั้นตอนที่ 10

- 11) คลิกขวาที่ปุ่ม Cancel เลือก Insert inside CommandButton-deleteEmployees →
ADF Faces Core → ReturnActionListener



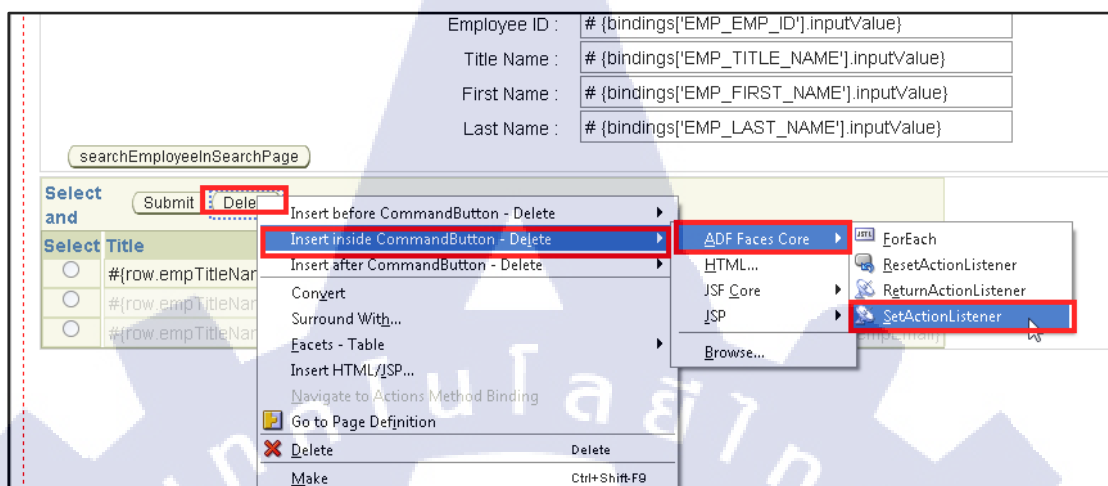
ภาพที่ 3.100 แสดงการสร้างหน้าลบข้อมูล ขั้นตอนที่ 11

- 12) สร้าง CommandButton ตั้งชื่อ Text:Delete , Action: dialog:delete



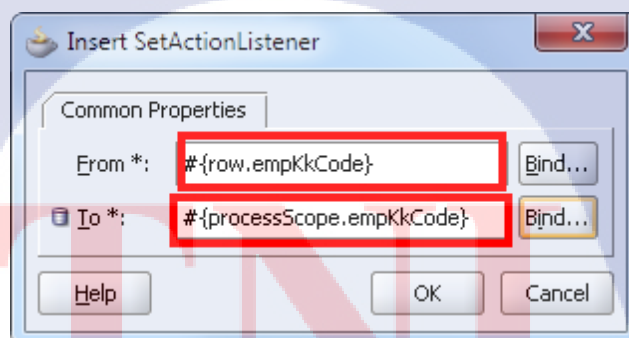
ภาพที่ 3.101 แสดงการสร้างหน้าลบข้อมูล ขั้นตอนที่ 12

13) คลิกขวาที่ปุ่ม Delete เลือก Insert inside CommandButton-Delete → ADF Faces Core → SetActionListener



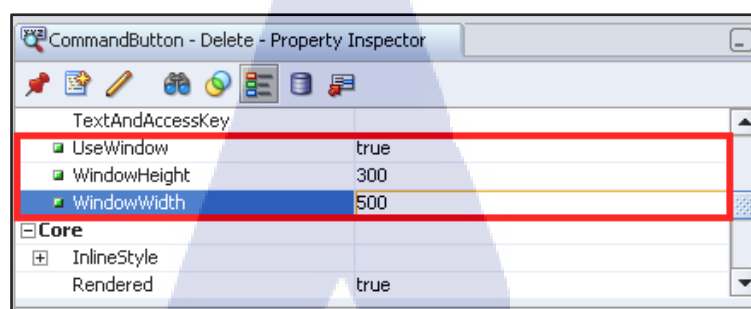
ภาพที่ 3.102 แสดงการสร้างหน้าลบข้อมูล ขั้นตอนที่ 13

14) กรอกข้อมูล จากนั้นกด OK



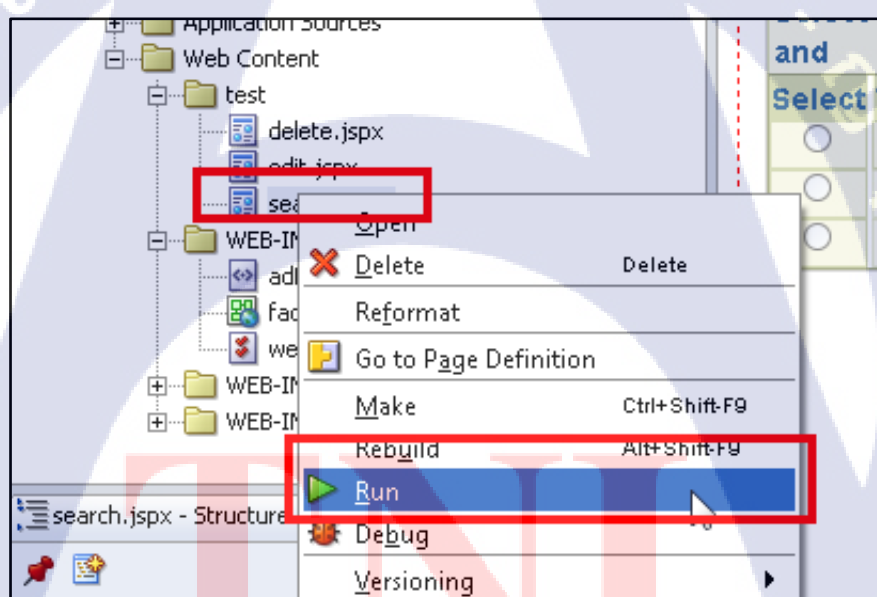
ภาพที่ 3.103 แสดงการสร้างหน้าลบข้อมูล ขั้นตอนที่ 14

15) ตั้งค่า Property ของปุ่ม Delete



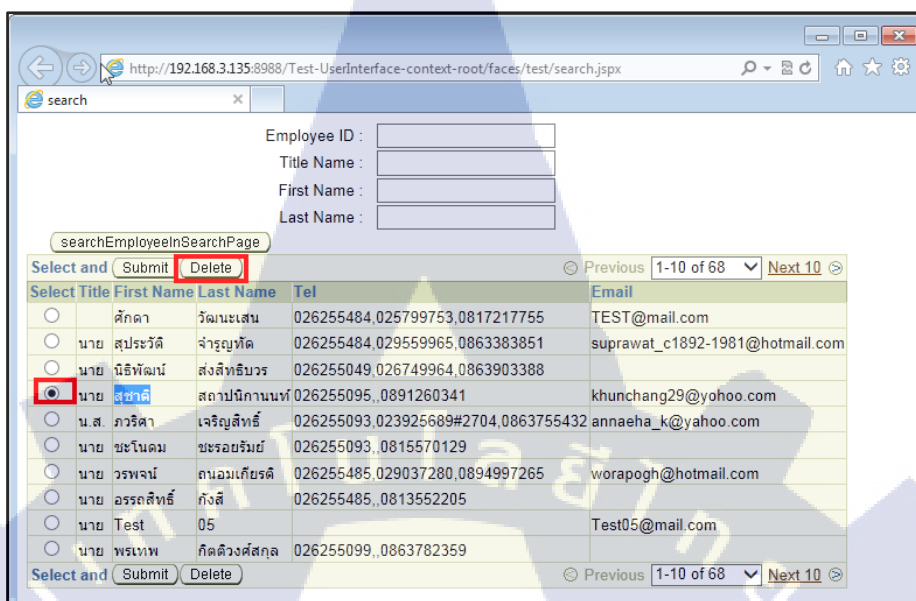
ภาพที่ 3.104 แสดงการสร้างหน้าต่างข้อมูล ขั้นตอนที่ 15

16) คลิกขวาที่ search.jspx ใน Navigator เลือก Run



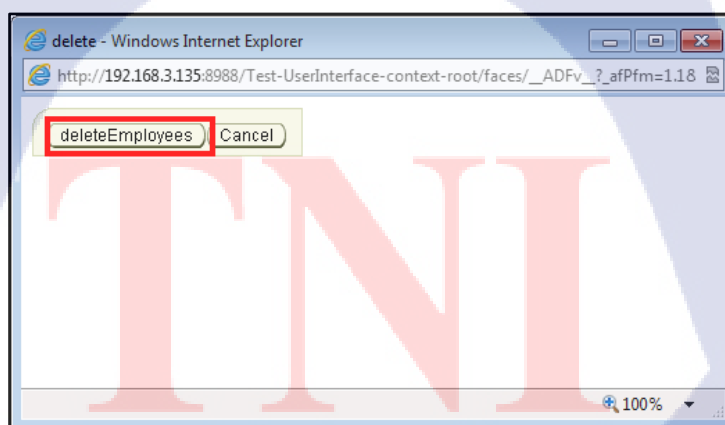
ภาพที่ 3.105 แสดงวิธีการรัน

17) เลือกข้อมูลที่ต้องการลบ จากนั้นกดปุ่ม Delete



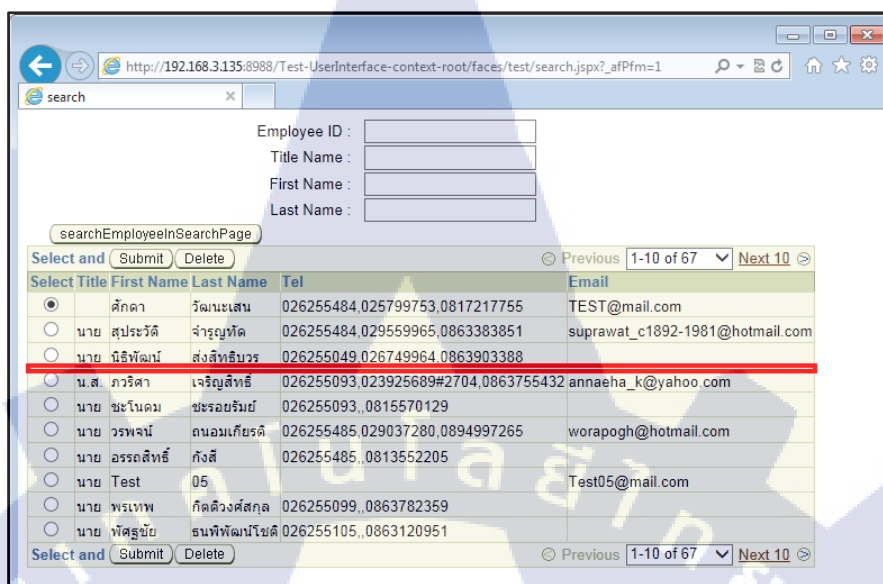
ภาพที่ 3.106 แสดงวิธีการลบข้อมูล ขั้นตอนที่ 1

18) Dialog จะเด้งขึ้นมา ให้เลือก DeleteEmployees



ภาพที่ 3.107 แสดงวิธีการลบข้อมูล ขั้นตอนที่ 2

19) กด searchEmployeeInSearchPage ข้อมูลที่ลบจะหายไป แสดงว่า ลบข้อมูลสำเร็จ



searchEmployeeInSearchPage

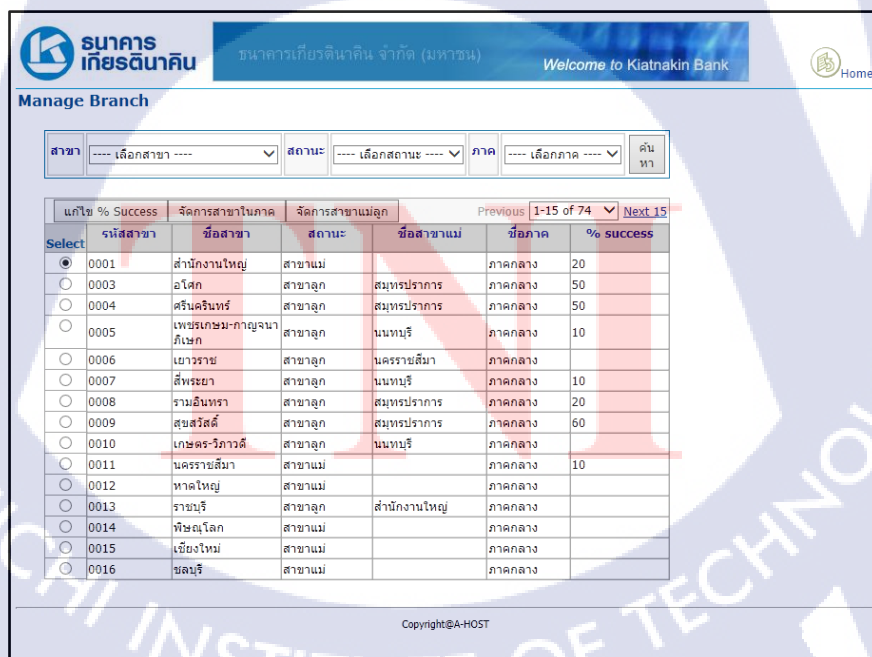
Select	Title	First Name	Last Name	Tel	Email
☒	ศักดิ์ดา	วัฒนะเสน		026255484,025799753,0817217755	TEST@mail.com
☐	นาย สุประวุฒิ	จำรูญโท		026255484,029559965,0863383851	suprawat_c1892-1981@hotmail.com
☐	นาย นิตยา	นิลวัฒน์	สงัดอินทร	026255049,026749964,0863903388	
☐	น.ส. กวรีดา	เจริญสิทธิ์		026255093,023925689#2704,0863755432	annaeha_k@yahoo.com
☐	นาย ชะโนดม	ชะรอยรัมย์		026255093,0815570129	
☐	นาย วรพจน์	ถนอมเกียรติ		026255485,029037280,0894997265	worapogh@hotmail.com
☐	นาย อรรถสิทธิ์	กังสี		026255485,0813552205	
☐	นาย Test	05			Test05@mail.com
☐	นาย พรเทพ	กิตติวงศ์สกุล		026255099,0863782359	
☐	นาย พิศุขชัย	ธนทิพัฒน์โชติ		026255105,0863120951	

ภาพที่ 3.108 แสดงวิธีการลบข้อมูล ขั้นตอนที่ 3

การจำลองหน้าค้นหาข้อมูลและหน้าแก้ไขข้อมูลด้วย Oracle ADF ตามที่ได้กล่าวมา เป็นส่วนหนึ่งของระบบเรจิดที่ได้สร้างจากพื้นฐานและองค์ประกอบเดียวกัน เพียงแต่เป็นการทำงานของหลายๆหน้ารวมกันเป็นระบบขึ้น



ภาพที่ 3.109 หน้าอินเทอร์เน็ตเฟสระบบเรจิด



ภาพที่ 3.110 หน้าอินเทอร์เน็ตเฟสจัดการสาขา

3.3.2 ความต้องการของลูกค้า (Change Request 25)

เป็นเอกสารแสดงความต้องการจากธนาคารเกียรตินาคิน เพื่อต้องการแก้ไขระบบให้สามารถรองรับการทำงานได้มากขึ้น ดังนั้น Change Request จะมีขึ้นมาเรื่อยๆซึ่งจะถูกจัดเป็นลำดับ ก่อนที่จะทำการแก้ไขจำเป็นต้องทราบถึงผลกระทบในการเพิ่มโค้ดหรือการเปลี่ยนแปลงที่เกิดขึ้น ทำให้เอกสารนี้มีความจำเป็นอย่างยิ่งในการสร้างแนวทางเบื้องต้นของการแก้ไขระบบ การทำงานจะเป็นขั้นตอน ใช้ระยะเวลาได้เหมาะสม

No.	Description	Impact / Solution	Priority	Category	Database	Portal & Other	Notes
25	Pursuit Change Request 25		66	66			
1	Design		6	6			
2	Application		2				
3	Database		2				
4	Package		2				
5	Table		2				
6	Package		2				
7	Table		2				
8	Table		2				
9	Table		2				
10	Table		2				
11	Table		2				
12	Table		2				
13	Table		2				
14	Table		2				
15	Table		2				
16	Table		2				
17	Table		2				
18	Table		2				
19	Table		2				
20	Table		2				
21	Table		2				
22	Table		2				
23	Table		2				
24	Table		2				
25	Table		2				
26	Table		2				
27	Table		2				
28	Table		2				
29	Table		2				
30	Table		2				
31	Table		2				
32	Table		2				
33	Table		2				
34	Table		2				
35	Table		2				
36	Table		2				
37	Table		2				
38	Table		2				
39	Table		2				
40	Table		2				
41	Table		2				
42	Table		2				
43	Table		2				
44	Table		2				
45	Table		2				
46	Table		2				
47	Table		2				
48	Table		2				
49	Table		2				
50	Table		2				
51	Table		2				
52	Table		2				
53	Table		2				
54	Table		2				
55	Table		2				
56	Table		2				
57	Table		2				
58	Table		2				
59	Table		2				
60	Table		2				
61	Table		2				
62	Table		2				
63	Table		2				
64	Table		2				
65	Table		2				
66	Table		2				
67	Table		2				
68	Table		2				
69	Table		2				
70	Table		2				
71	Table		2				
72	Table		2				
73	Table		2				
74	Table		2				
75	Table		2				
76	Table		2				
77	Table		2				
78	Table		2				
79	Table		2				
80	Table		2				
81	Table		2				
82	Table		2				
83	Table		2				
84	Table		2				
85	Table		2				
86	Table		2				
87	Table		2				
88	Table		2				
89	Table		2				
90	Table		2				
91	Table		2				
92	Table		2				
93	Table		2				
94	Table		2				
95	Table		2				
96	Table		2				
97	Table		2				
98	Table		2				
99	Table		2				
100	Table		2				

ภาพที่ 3.111 ตัวอย่างเอกสาร Change Request 25

บทที่ 4

ผลการดำเนินงาน การวิเคราะห์และสรุปผลต่างๆ

4.1 ขั้นตอนและผลการดำเนินงาน

จากการที่ได้ไปสหกิจ ณ บริษัท A-HOST Co.,Ltd เป็นเวลาประมาณ 4 เดือน ตั้งแต่วันที่ 3 มิถุนายน พ.ศ. 2556 ถึงวันที่ 3 ตุลาคม พ.ศ. 2556 ซึ่งในการสหกิจครั้งนี้ ได้รับมอบหมายหน้าที่ในตำแหน่ง Programmer เพื่อช่วยสนับสนุนโครงการของธนาคารเกียรตินาคินตามส่วนที่ได้รับมอบหมาย และทำหน้าที่อื่นๆนอกเหนือจากนี้ด้วย ซึ่งทำให้ได้รับความรู้ทั้งด้านการเขียนโปรแกรม ทักษะในการประกอบอาชีพ ซึ่งสามารถนำไปใช้ในการทำงานและชีวิตประจำวันได้เป็นอย่างดี โดยมีงานดังนี้

4.1.1 ศึกษา HPAOL/Pursuit และทำตาม

4.1.1.1. ขั้นตอนการดำเนินงาน

- 1) ศึกษากระบวนการทำงานของระบบ HPAOL(ระบบเช่าซื้อรถยนต์) และระบบ Pursuit(ระบบเร่รัด) ของธนาคารเกียรตินาคิน
- 2) ทำเอกสาร Testcase ของระบบที่ศึกษาเพื่อทำความเข้าใจในส่วนต่างๆมากขึ้น
- 3) ทำเอกสาร Manual ของระบบ Pursuit ในปัจจุบัน ก่อน Change Request 25 เพื่อศึกษาว่าระบบ Pursuit ในปัจจุบันทำงานอะไรได้บ้าง
- 4) สร้างระบบขึ้นมา ใหม่โดยเลียนแบบจาก ระบบเดิม เพื่อฝึกฝนและทำความเข้าใจระบบ
- 5) ทำ Testcase ของระบบที่เราทำขึ้นมาใหม่เพื่อตรวจสอบความเรียบร้อย

4.1.1.2. ผลการดำเนินงาน

จากการศึกษาระบบ HPAOL/Pursuit และทำตามโดยใช้โปรแกรม Oracle JDeveloper ด้วยการใช้ Application Development Framework(ADF) ในการศึกษาทำให้เรียนรู้การเขียนโปรแกรมด้วย Java โดยประยุกต์ใช้ร่วมกับ SQL ทำให้สามารถเข้าใจและเรียนรู้ระบบเพื่อนำไปใช้ในการพัฒนาระบบต่อไปได้ง่ายขึ้น

4.1.2 ศึกษาวิธีการ export PDF จาก เว็บไซต์

4.1.2.1. ขั้นตอนการดำเนินงาน

- 1) ค้นหา และศึกษาข้อมูลจากอินเทอร์เน็ต
- 2) นำข้อมูลและตัวอย่างโปรแกรมมาทดลอง
- 3) ปรึกษาพี่เลี้ยงและได้รับคำแนะนำจากพี่เลี้ยง
- 4) สรุปผลการศึกษาเป็นเอกสาร

4.1.2.2. ผลการดำเนินงาน

ได้ ศึกษาทำความเข้าใจกับ ข้อมูลและ โค้ดที่หาได้และจากที่พี่เลี้ยง ได้แนะนำมาให้ศึกษา และ หลังจากทดลองเขียนโปรแกรม ด้วยเครื่องมือต่างๆที่หาได้แล้วพบว่าสามารถทำได้ แต่มีปัญหาด้านการใช้ภาษาอื่นนอกจากภาษาอังกฤษซึ่งมีความยุ่งยากมากในการใช้ในภาษาอื่นๆ

4.1.3 แก้ไขระบบงานเร่งรัดของธนาคารเกียรตินาคิน ในส่วน ระบบสาขาแม่ลูก ตาม และการค้นหางานจากภาคตาม Change Request 25

4.1.3.1 ขั้นตอนการดำเนินงาน

- 1) เพิ่ม Table ใหม่และแก้ไข Table เดิมในฐานข้อมูลโดยใช้โปรแกรม Toad for Oracle 10
- 2) เปลี่ยนระบบ สาขาแม่ลูกเดิมให้เป็นแบบ Top-Link
- 3) เพิ่ม และแก้ไข SessionBean ที่ใช้งานให้เหมาะสมกับการใช้งาน

- 4) เพิ่มหน้าในการทำงานเรื่องการเพิ่ม แก้ไขค่า เปอร์เซ็นความสำเร็จ
- 5) เพิ่มหน้าการกำหนดภาคให้สาขาแม่ลูก
- 6) ทำเอกสาร Testcase ในส่วนสาขาแม่ลูก
- 7) แก้ไขหน้าค้นหางานโดยเพิ่มเงื่อนไขภาคลงไปในการค้นหาได้
- 8) ทำเอกสาร Testcase ในส่วนที่แก้ไขใหม่

4.1.3.2 ผลการดำเนินงาน

ได้ศึกษาและพัฒนาโค้ดด้วยตนเอง ซึ่งทำให้เรามีประสบการณ์และความรู้ใน การแก้ไข และการเขียนโปรแกรมมากขึ้นนอกจากนี้ยังได้ ทราบถึง ข้อผิดพลาดของตนเอง และแนวทางแก้ไขข้อผิดพลาดนั้น

4.1.4 ศึกษาโปรแกรม OpenKM

4.1.4.1 ขั้นตอนการดำเนินงาน

- 1) ศึกษาโปรแกรม OpenKM จากอินเทอร์เน็ต
- 2) ลองทำการติดตั้งโปรแกรม OpenKM ในVersion ต่างๆ
- 3) แก้ไขการตั้งค่าการติดตั้งให้ใช้งานได้
- 4) จัดทำเอกสารคู่มือการติดตั้ง (Manual) โปรแกรม OpenKM
- 5) ตรวจสอบและแก้ไขเอกสารคู่มือการติดตั้ง

4.1.4.2 ผลการดำเนินงาน

ได้รับความรู้เกี่ยวกับการ ติดตั้งใช้งานและการแก้ไขการตั้งค่า โปรแกรม OpenKM ให้สามารถใช้งานได้ และได้ฝึกทำเอกสารคู่มือการติดตั้งเพื่อให้บุคคลอื่นในองค์กร นำไปติดตั้งใช้เพื่อใช้งานได้ง่าย

4.2 ผลการวิเคราะห์ข้อมูล

ในการปฏิบัติงานได้รับมอบหมายงานระบบเร่งรัดเป็นหลัก ดังนี้

สำหรับการพัฒนาระบบเร่งรัดซึ่งเป็นงานหลักที่ช่วยสนับสนุนความต้องการของลูกค้าก็คือ ธนาคารเกียรตินาคิน โดยระบบนี้มีส่วนช่วยการทำงานของระบบเช่าซื้อรถยนต์ของธนาคาร ซึ่งช่วยให้การทำงานรวดเร็วในการจัดการเอกสารต่างๆ มากขึ้น เปรียบเทียบกับการทำงานโดยไม่มีระบบแล้ว ในยุคที่ยังไม่มีระบบอินเทอร์เน็ตนั้น ทุกอย่างต้องถูกจัดเก็บเป็นเอกสาร ทำให้การค้นหา หรือ แก้ไขทำได้ช้า

4.3 วิจัยข้อมูลโดยเปรียบเทียบผลที่ได้รับกับวัตถุประสงค์และจุดมุ่งหมายการปฏิบัติงานหรือการจัดทำโครงการ

เมื่อทำการเปรียบเทียบงานที่ได้รับมอบหมายก่อนหน้าปฏิบัติงานรวมถึงเป้าหมายก่อนเสร็จสิ้นการสหกิจศึกษานั้น สามารถสรุปผลได้ดังนี้

- 1) การใช้โปรแกรม JDeveloper เพื่อใช้พัฒนา Oracle ADF เป็นส่วนเพิ่มเติมจากระบบเบื้องต้น เมื่อเข้าใจระบบการทำงานของ Oracle ADF ก็สามารถใช้โปรแกรมได้อย่างรวดเร็ว
- 2) ความรู้เบื้องต้นเกี่ยวกับภาษา SQL ,PL/SQL ,JAVA สามารถนำมาใช้ในการปฏิบัติงาน
- 3) การปฏิบัติงาน Testing ระบบเร่งรัดของธนาคารเกียรตินาคิน สามารถทำได้ตามเป้าหมายที่ได้รับมอบหมายไว้
- 4) การช่วยสนับสนุนในการทำ HPAOL / Pursuit Change Request เป็นไปตามที่วางไว้
- 5) การพัฒนา Pursuit Change Request 25 สามารถทำตามที่วางไว้ได้เกือบทั้งหมด เหลือเพียงบางส่วนเท่านั้น

บทที่ 5

บทสรุปและข้อเสนอแนะ

5.1 สรุปผลการดำเนินงาน

จากการที่ได้ทำงานกับบริษัท A-Host จำกัดเป็นเวลา 4 เดือน ทำให้ได้รับความรู้และทักษะใหม่ๆ เรียนรู้วิธีและเรียนรู้ความต้องการทางสาย IT ว่ามีหนทางใดในการให้บริการได้บ้าง ได้เรียนรู้เทคโนโลยีใหม่ๆ ทั้ง ภาษาในการพัฒนาและ Framework ที่ช่วยให้พัฒนาได้เร็วขึ้นและมีประสิทธิภาพมีความยืดหยุ่นในการพัฒนาในอนาคตและการทำให้ Application ที่พัฒนาทำงานบน Server ยังไงให้ผู้ใช้งานนำไปใช้บริการ ได้รู้จัก Server ที่ใช้ทำงานต่างๆ รวมถึง Software ที่ใช้และให้บริการหลากหลายรูปแบบทั้งของ Oracle และได้นำ Software บางตัวมาศึกษาใช้งาน และเรียนรู้ข้อควรปฏิบัติในการทำงาน ได้เพิ่มทักษะวิชาพื้นฐานจากปัญหาที่เกิดขึ้นในการทำงาน และได้แนวทางการแก้ปัญหาจากพี่ที่ให้อำนาจปรึกษา

5.2 แนวทางการแก้ไข้ปัญหา

ในเวลาที่มีปัญหาในการทำงานได้ศึกษาหาทางแก้ไขด้วยตัวเองก่อนจะไปถามพี่ผู้ดูแล แต่ถ้าปัญหานั้นใช้เวลาในการหาจนเกินไปควรจะปรึกษาพี่เพื่อไม่ให้เกิดความล่าช้าแต่ก็ไม่ใช่ว่าไปถามบ่อยจนรบกวนเวลาทำงานพี่ผู้ดูแลมากเกินไป การทำงานนั้นทำงานกันเป็นทีมดังนั้นเวลาทำงานงานของเราไม่ควรไปกระทบกับเวลาทำงานของคนอื่นโดยอะไรที่อยู่นอกเหนือความรับผิดชอบควรสอบถามหรือขออนุญาตก่อนลงมือปฏิบัติจะได้ไม่เกิดปัญหาอื่นตามมา ส่วนเรื่องการหาเนื้อหาเพิ่มเติมที่เป็นสิ่งใหม่ถ้าต้องการจะเข้าไปหาในแหล่งความรู้ของเทคโนโลยีนั้นจริงๆ เสียก่อนเพราะเป็นแหล่งความรู้ที่น่าเชื่อถือได้มากที่สุด ถ้าหากต้องการหารายละเอียดพิเศษอื่นๆ นั้น เมื่อหามาได้ควรจะทดลองด้วยตัวเองเสียก่อนเพื่อความถูกต้อง และชัดเจนของข้อมูล

5.3 ข้อเสนอแนะจากการฝึกงาน

การศึกษารายวิชาที่ไปทำสหกิจว่าทางบริษัทใช้เทคโนโลยีอะไรเป็นหลักในการให้บริการ และศึกษาพื้นฐานด้านนั้นให้เป็นอย่างดีเมื่อถึงเวลาจริงเราไปศึกษาพื้นฐานใหม่จะใช้เวลาค่อนข้างนาน เมื่อเข้าไปก็ต้องรับฟังความคิดเห็นและแนวทาง แล้ววิเคราะห์นำมาประยุกต์ใช้พยายามจับใจความสิ่งที่ต้องการจะสื่อสาร ประพฤติให้ถูกกาลเทศะและสุภาพแต่ไม่ทำให้สถานการณ์ตึงเครียดเกินไป รักในงานที่ทำหรือเปิดใจลองทำเพื่อความรู้และทักษะใหม่ๆ ตั้งใจทำงานให้ได้

ผลงานที่มีคุณภาพด้วยการตรวจสอบงานอย่างถี่ถ้วนไม่ให้เกิดข้อผิดพลาด วางแผนงานล่วงหน้า และจัดระเบียบการมีข้อมูลอยู่กับมือทำให้บริหารงานที่กำลังเข้ามาได้ และรู้จักตัดสินใจในการแก้ปัญหาเฉพาะหน้าได้



เอกสารอ้างอิง

1. Jeff Gallus, Gary Williams, Kate Heap, 2006, “Developing a Search Page”, **Oracle Application Development Framework Tutorial 10g Release 3 (10.1.3)**., pp. 85-98.
2. Jeff Gallus, Gary Williams, Kate Heap, 2006, “Developing an Edit Page”, **Oracle Application Development Framework Tutorial 10g Release 3 (10.1.3)**., pp. 141-153.
3. A-HOST, 2007, “Pursuit Project Introduction”, **Pursuit Project Introduction**, pp. 2-15.
4. A-HOST, 2008, “Project: HPAOL-Pursuit Module”, **User Interface Description**, pp. 8-179.
5. A-HOST, 2008, “คู่มือการใช้งาน LookUp Table”, **QuickCode Manual**, pp. 1-8.

ภาคผนวก ก.

TNI

ภาพถ่ายการปฏิบัติงานสหกิจ ณ บริษัท A-HOST Co.,Ltd.



ภาพที่ ก.1 การปฏิบัติงานขณะอยู่ออฟฟิศ ณ บริษัท A-HOST



ภาพที่ ก.2 การ Training Oracle ADF 11g



ภาพที่ ก.3 หมู่ประจำรุ่น ของการสหกิจศึกษา



ภาพที่ ก.4 สำนักงานใหญ่ธนาคารเกียรตินาคิน สาขาอโศก

ภาคผนวก ข.

TNI

ตัวอย่าง SQL, PL/SQL ที่ปรับปรุงโดยใช้ Toad ในการจัดการในฐานข้อมูล

Function FUNC_DATE_ASSIGN

```
CREATE OR REPLACE FUNCTION PS.FUNC_DATE_ASSIGN (CONTACT_NO in
VARCHAR2,PURSUIT_NO in VARCHAR2) RETURN VARCHAR2
```

```
IS
```

```
tmpDate VARCHAR2(30 byte);
```

```
BEGIN
```

```
select to_char((select CREATED_DATE from JOB_PURSUIT_INFO where
JPI_PURSUIT_NO = PURSUIT_NO and JPI_CONTRACT_NO = CONTACT_NO ) +
QC_LOOKUP_END,'dd/mm/yyyy') into tmpDate
```

```
from QUICK_CODES
```

```
WHERE QC_LOOKUP_TYPE = 'SYS_WORK_FLOW' AND QC_LOOKUP_CODE = '1';
```

```
RETURN tmpDate;
```

```
EXCEPTION
```

```
WHEN OTHERS THEN
```

```
null;
```

```
END FUNC_DATE_ASSIGN;
```

```
/
```

View BRANCH_HIE_VIEW

```
DROP VIEW PS.BRANCH_HIE_VIEW;
```

```
CREATE OR REPLACE FORCE VIEW PS.BRANCH_HIE_VIEW
```

```

(
    B_BRANCH_CODE,
    B_NAME,
    B_PARENT_CODE,
    B_MANAGER_ID,
    BH_PARENT,
    PARENT_NM,
    MAXCHILD,
    HIE_STATUS,
    HIE_STATUS_DESC,
    B_REGION_ID,
    B_REGION_NAME,
    B_SUCCESS
)
AS
    SELECT bv1.B_BRANCH_CODE,
           bv1.B_NAME,
           bv1.B_PARENT_CODE,
           bv1.B_MANAGER_ID,
           b1.BH_PARENT,
           bv2.B_NAME PARENT_NM,
           b2.maxChild,
           CASE
               WHEN b2.maxChild IS NULL
               THEN
                   CASE WHEN b1.BH_PARENT IS NULL THEN '-' ELSE 'C' END
               ELSE
                   'P'
           END
           hie_status,

```



```

CASE
  WHEN b2.maxChild IS NULL
  THEN
    CASE
      WHEN b1.BH_PARENT IS NULL
      THEN
        'ไม่ได้ระบุ'
      ELSE
        'สาขาถูก'
      END
    ELSE
      'สาขาแม่'
    END
  hie_status_desc,
CASE
  WHEN b2.maxChild IS NULL
  THEN
    CASE WHEN b1.BH_PARENT IS NULL THEN '-' ELSE B1.BH_REGION END
  ELSE
    B3.BH_REGION
  END
  B_REGION_ID,
CASE
  WHEN b2.maxChild IS NULL
  THEN
    CASE
      WHEN b1.BH_PARENT IS NULL THEN '-'
      ELSE MR.REGION_NAME
    END
  ELSE

```



```

        MR.REGION_NAME
    END

    REGION_NAME,
    QC.QC_LOOKUP_VALUE1 SUCCESS
FROM BRANCH_V bv1
LEFT JOIN BRANCH_HIERARCHY b1
    ON bv1.b_branch_code = b1.bh_child
LEFT JOIN ( SELECT bh_parent, MAX (bh_child) maxChild
            FROM BRANCH_HIERARCHY
            GROUP BY bh_parent) b2
    ON bv1.b_branch_code = b2.bh_parent
LEFT JOIN (SELECT DISTINCT (BH_PARENT) b_parent, BH_REGION
            FROM BRANCH_HIERARCHY) b3
    ON BV1.B_BRANCH_CODE = B3.b_parent
LEFT JOIN BRANCH_V bv2
    ON b1.BH_PARENT = bv2.b_branch_code
LEFT JOIN MASTER_REGION mr
    ON B1.BH_REGION = MR.REGION_ID OR B3.BH_REGION = MR.REGION_ID
LEFT JOIN QUICK_CODES qc
    ON BV1.B_BRANCH_CODE = QC.QC_LOOKUP_CODE
    AND QC.QC_LOOKUP_TYPE = 'LV_SUCCESS';

```

ประวัติผู้จัดทำโครงการ



ชื่อ – นามสกุล

นายเฉลิมชัย อ่องจรรย์

วัน เดือน ปีเกิด

29 พฤศจิกายน 2534

ประวัติการศึกษา

ระดับประถมศึกษา โรงเรียนอนุบาลสมุทรสาคร พ.ศ. 2546

ระดับมัธยมศึกษา โรงเรียนสมุทรสาครวิทยาลัย พ.ศ. 2552

ระดับอุดมศึกษา คณะเทคโนโลยีสารสนเทศ สาขาเทคโนโลยีมัลติมีเดีย พ.ศ. 2556

สถาบันเทคโนโลยีไทย-ญี่ปุ่น

ทุนการศึกษา 1. ทุนสนับสนุนการศึกษา พ.ศ. 2554

ประวัติการฝึกอบรวม 1

. การทำอนิเมชั่น ณ บริษัท Imagimax Studio จำกัด

2.

Creative Web Design & Development ณ บริษัท Asiasoft

3. Training Pre-Cooperative Education โครงการสหกิจฯ บริษัท A-HOST

Co., Ltd.

ผลงานที่ได้รับการตีพิมพ์ - ไม่มี -