

การสร้าง **Data Model** ด้วย โปรแกรม **DB Designer**

204222 FUND OF DATABASE SYSTEMS
Computer Science Department
Faculty of Science, Chiang Mai University

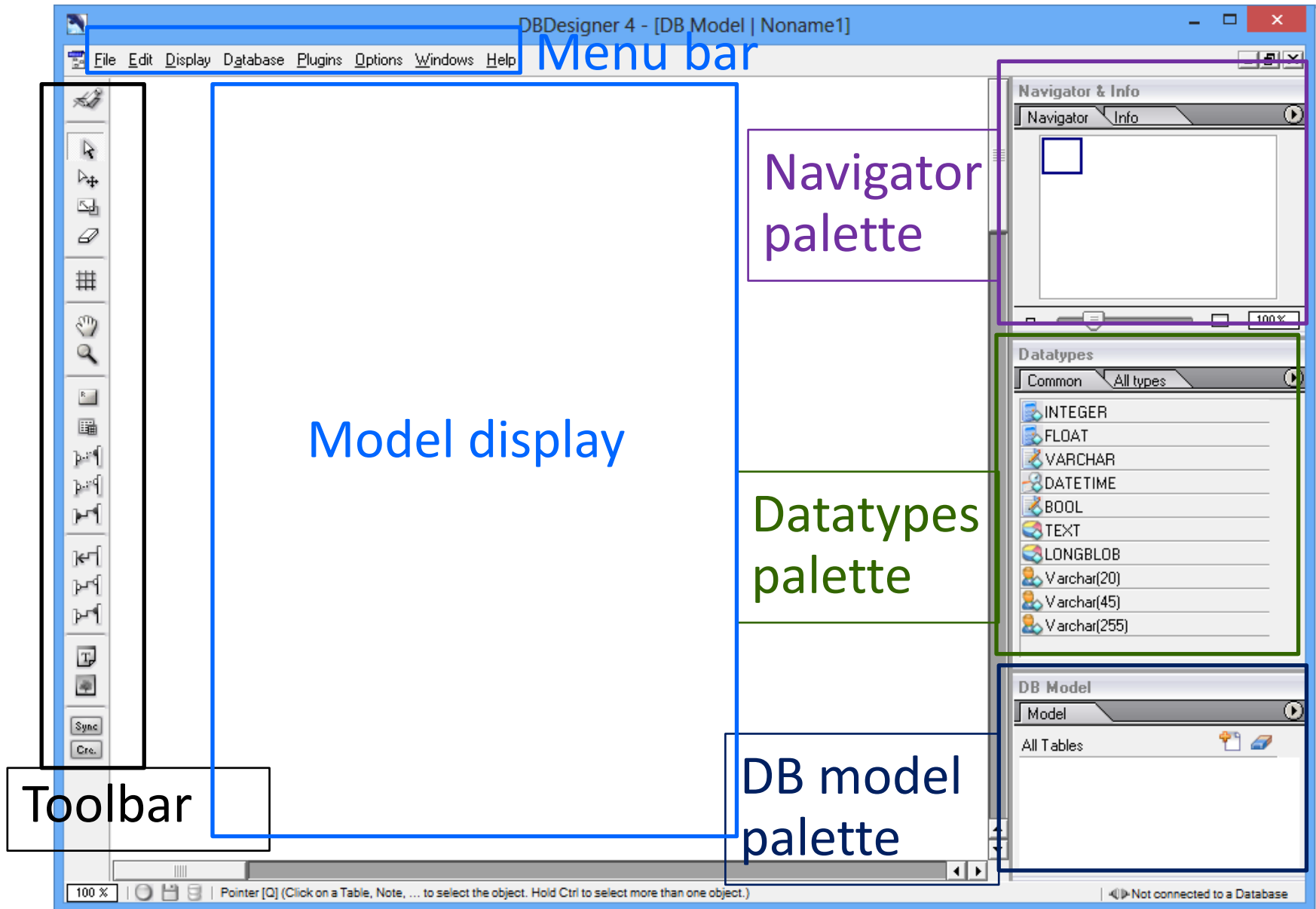
Asst. Prof. Dr. Churee Techawut

Additional reference: <http://fabforce.net/downloads.php>

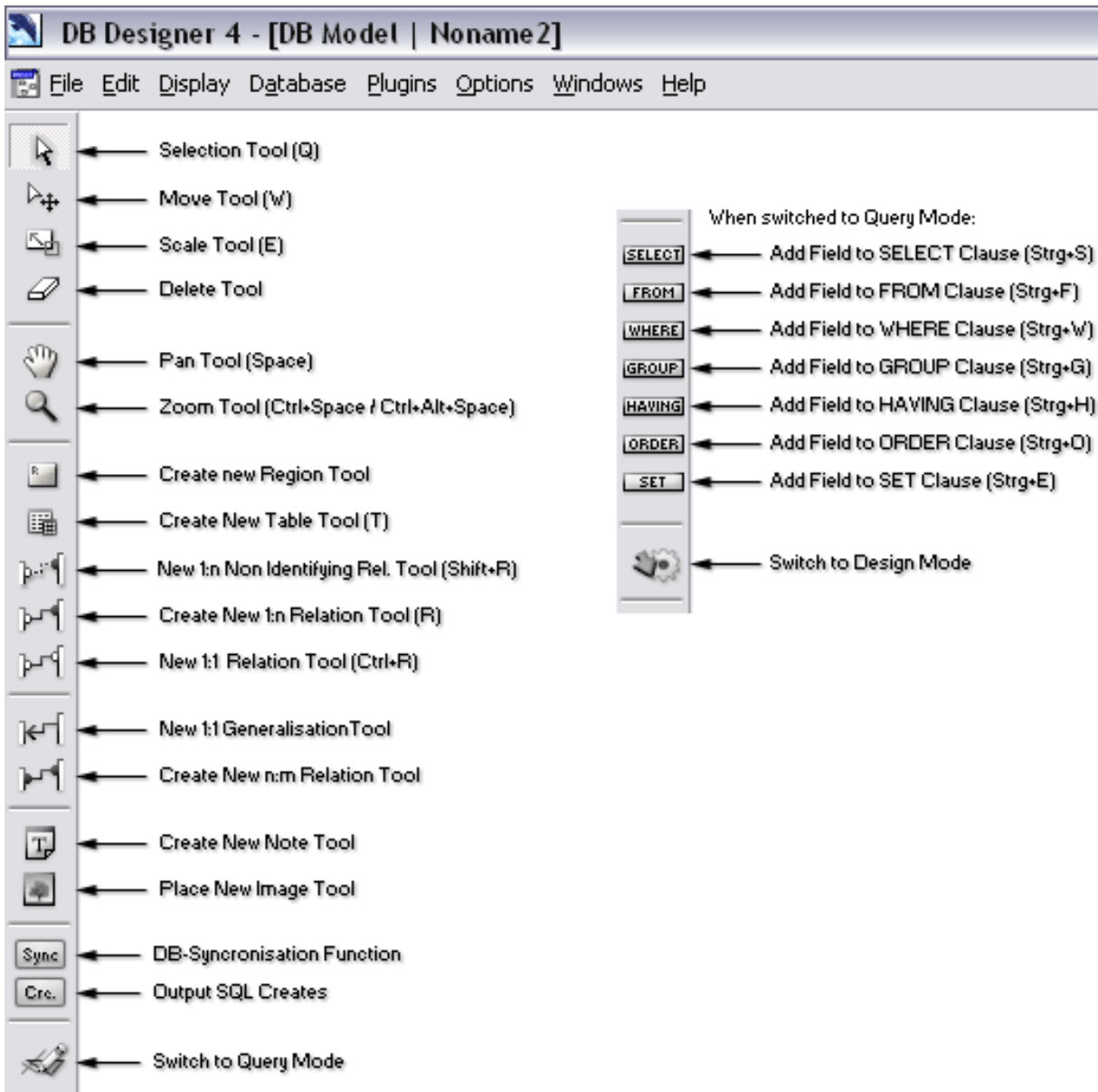
Outline

1. ส่วนประกอบหน้าจอ
2. การสร้าง ER Model อย่างง่าย
3. Export data model to the SQL script file
4. การสร้างพจนานุกรมข้อมูล (Data Dictionary)
5. แบบฝึกหัด

1. ส่วนประกอบหน้าจอ

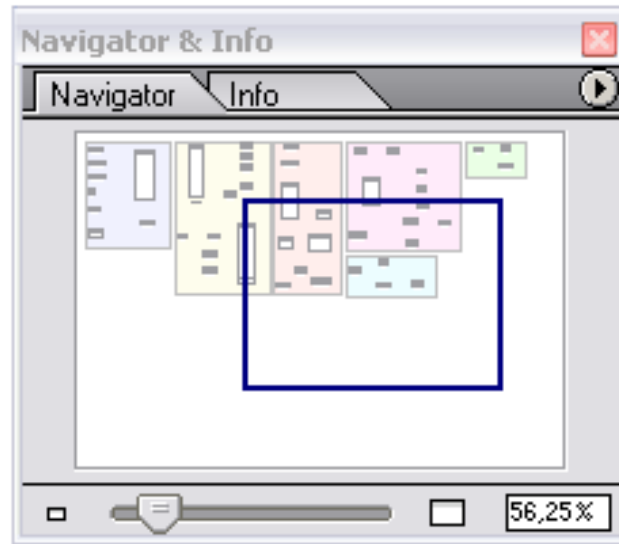


1.1. Toolbar



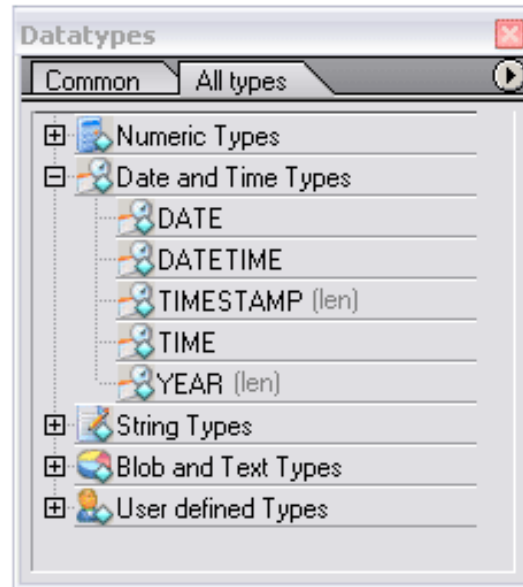
แถบเครื่องมือ ทำให้
สามารถเข้าถึง
เครื่องมือสำคัญได้
อย่างรวดเร็ว

1.2. Navigator palette



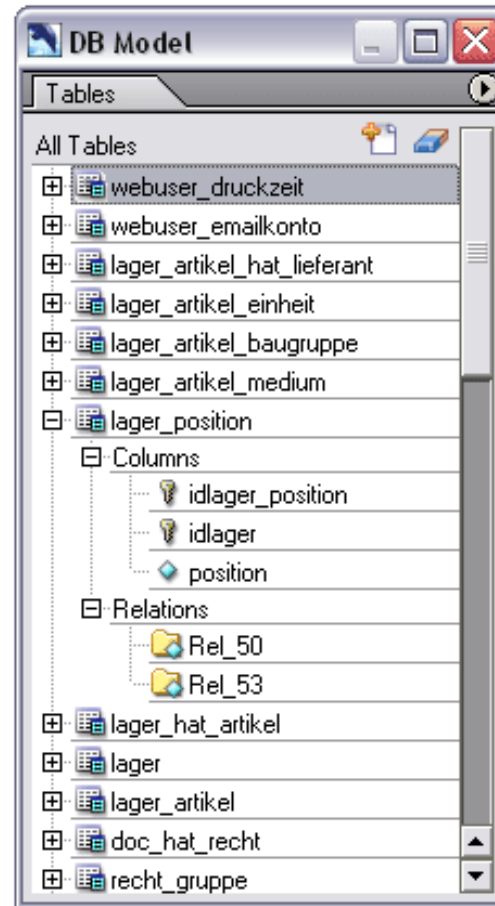
แสดงพื้นที่ของ Model ที่กำลังทำงาน ณ ปัจจุบัน

1.3. Data type palette



แสดงรายการ Data types ที่สามารถถูกเลือกมานิยาม (หรือ กำหนดโครงสร้าง) ฐานข้อมูล

1.4. DB model palette



แสดงตารางทั้งหมดที่ปรากฏใน Model รวมถึง Columns และ Relations

2. การสร้าง ER model อย่างง่าย

2.1. สร้าง Table / Entity

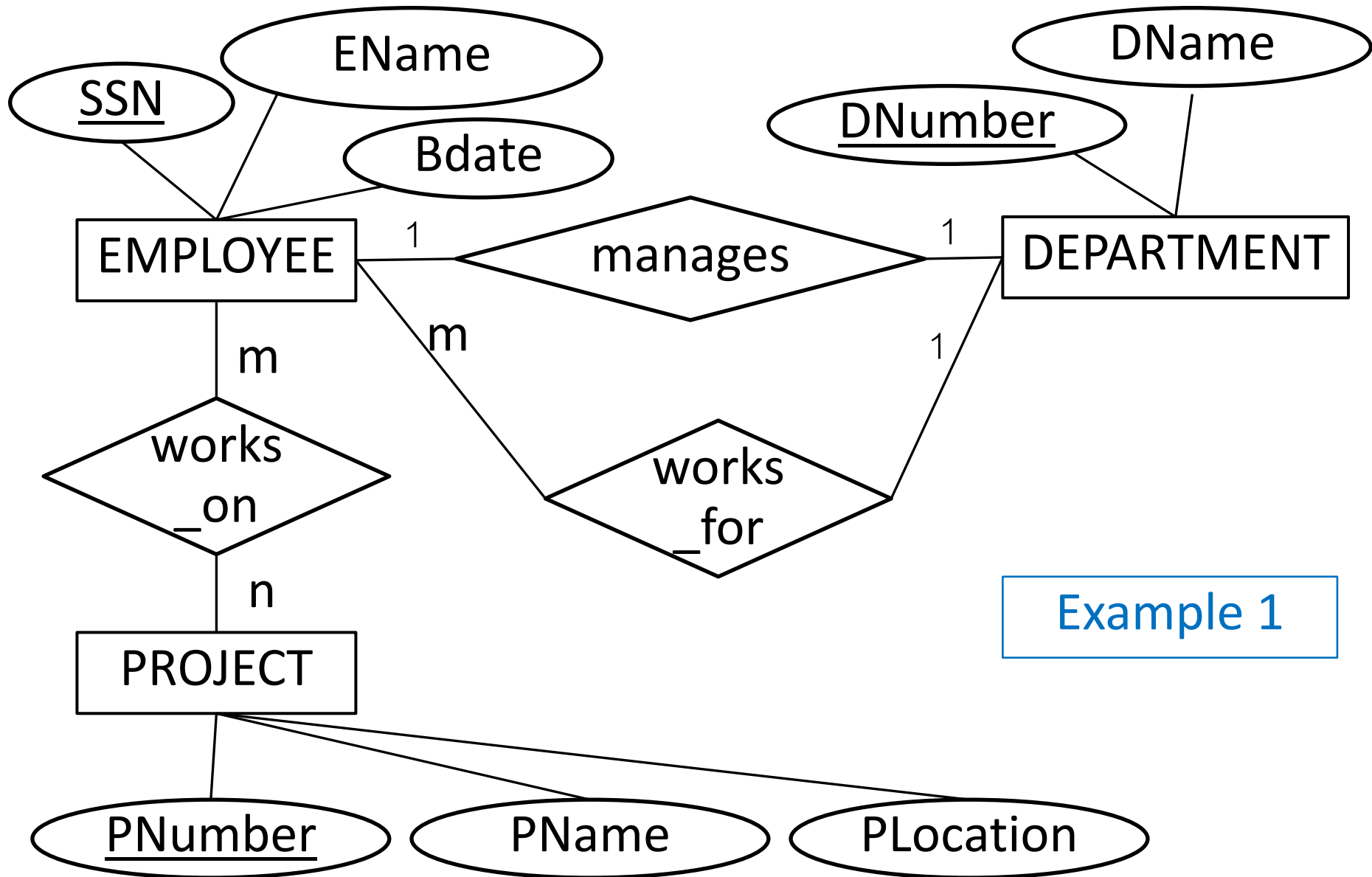
2.2. สร้าง Relationship

2.2.1. สร้าง 1 : 1 Relationship

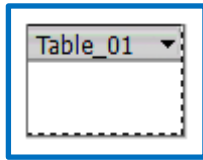
2.2.2. สร้าง 1 : n Relationship

2.2.3. สร้าง m : n Relationship

2. การสร้าง ER model อย่างง่าย



2.1. สร้าง Table / Entity



2. ดับเบิ้ลคลิกที่ **New Table**



New Table [T] (Click on the EER-Model or a region to create a new Table.)

1. เลือกเครื่องมือสร้าง **Table**

2.1. สร้าง Table / Entity

1. ใส่ชื่อ Table

2. ระบุประเภท Table

3. ระบุชื่อ Columns / Attributes

4. กำหนด Data types

5. กำหนด Primary key

6. คลิก OK

Column Name	DataType	NN	AI	Flags	Default Value	Comments
SSN	CHAR(13)	✓		<input type="checkbox"/> BINARY		หมายเลขบัตรประชาชน
EName	VARCHAR(30)	✓		<input type="checkbox"/> BINARY		ชื่อพนักงาน
Bdate	DATE	✓		<input type="checkbox"/> BINARY		วันเกิด

2.1. สร้าง Table / Entity

ให้ศ.สร้าง Table DEPARTMENT ด้วยตนเอง กำหนดรายละเอียดตารางดังนี้

The screenshot shows the 'Table Editor' window with the following configuration:

- Table Name:** DEPARTMENT
- Table Prefix:** Default (no prefix)
- Table Type:** InnoDB (Transaction-safe tables with row loc)
- Weak entity:** is n:m Table

Column Name	Data Type	NN	AI	Flags	Default Value	Comments
DNumber	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/> UNSIGNED <input type="checkbox"/> ZEROFILL		รหัสพนักงาน
DName	VARCHAR(25)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/> BINARY		ชื่อพนักงาน

Indices:

- Indexname:** PRIMARY
- Index Type:** PRIMARY
- Columns:** DNumber

2.1. สร้าง Table / Entity

ให้หนศ.สร้าง Table PROJECT ด้วยตนเอง กำหนดรายละเอียดตารางดังนี้

The screenshot shows the 'Table Editor' window with the following configuration:

- Table Name: PROJECT
- Table Prefix: Default (no prefix)
- Table Type: InnoDB (Transaction-safe tables with row loc)
- Weak entity: is n:m Table

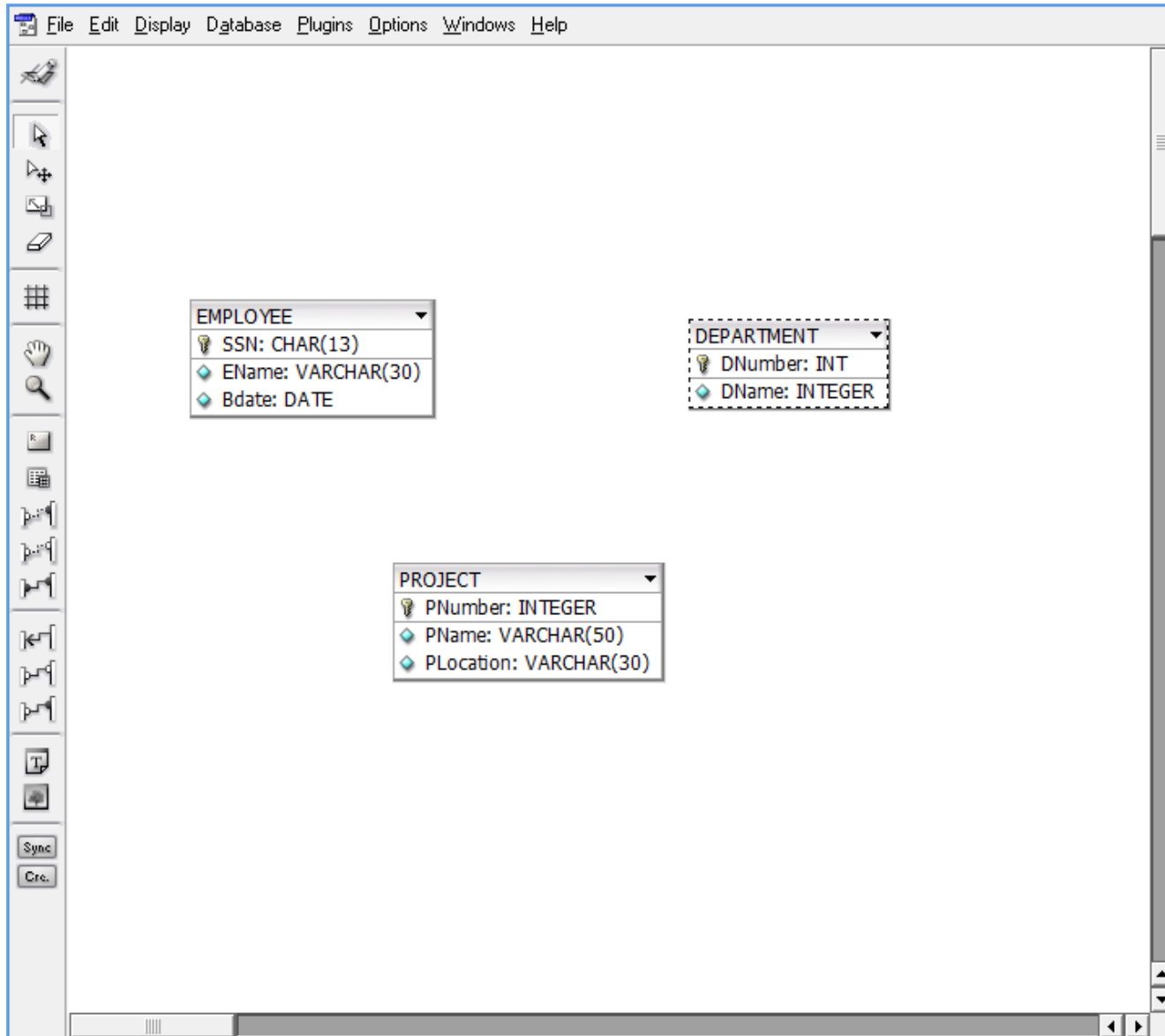
Column Name	Data Type	NN	AI	Flags	Default Value	Comments
PNumber	INTEGER	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> UNSIGNED <input type="checkbox"/> ZEROFILL		รหัสโครงการ
PName	VARCHAR(50)			<input type="checkbox"/> BINARY		ชื่อโครงการ
PLocation	VARCHAR(30)			<input type="checkbox"/> BINARY		สถานที่จัดทำโครงการ

Indices

- Table Options
 - Advanced
 - Standard Inserts
 - Comments
- PRIMARY
 - Indexname: PRIMARY
 - Index Type: PRIMARY
 - Columns (use Drag'n'Drop to add Columns): PNumber

2.1. สร้าง Table / Entity

จะได้ Table EMPLOYEE, DEPARTMENT, และ PROJECT ดังรูป



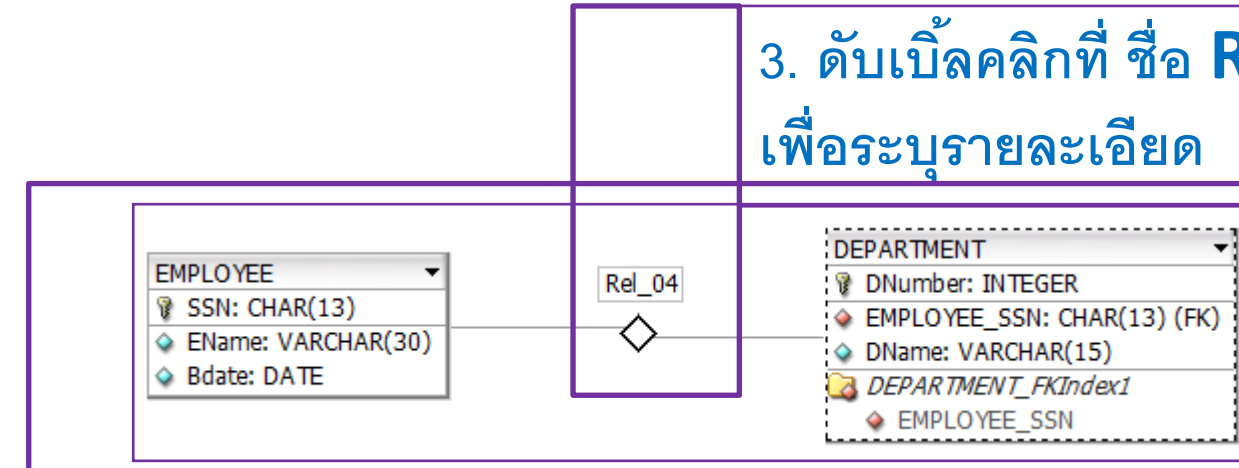
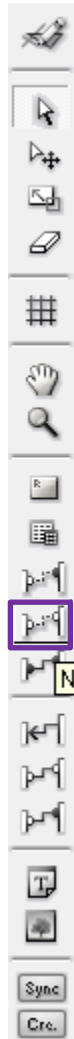
2.2. สร้าง Relationship

มีขั้นตอนดังนี้

1. เลือกเครื่องมือสร้าง Relationship
2. คลิก Source table (การคลิกเลือก Table ครั้งแรก)
3. คลิก Destination table (การคลิกเลือก Table ครั้งที่สอง)
4. ดับเบิลคลิกที่ Relationship และกำหนดรายละเอียดใน Relation editor

****หมายเหตุ: การเลือก Table ใดๆให้เป็น Destination table คือการกำหนดว่า ให้นำ Primary key ของ Source table มาเป็น Foreign key ใน Destination table ****

2.2.1. สร้าง 1 : 1 Relationship ชื่อ manages



3. ดับเบิ้ลคลิกที่ ชื่อ **Relationship** เพื่อระบายละเอียด

2. คลิกที่ **Table EMPLOYEE** และคลิกที่ **Table DEPARTMENT**

1. เลือกเครื่องมือสร้าง **1 : 1 Relationship**

New 1:1 Non-Identifying-Relation [] (Click on one table, then on the other table to link these two tables by a 1:1 relation.)

2.2.1. สร้าง 1 : 1 Relationship ชื่อ manages

1. กำหนดชื่อ
Relationship

2. ตรวจสอบและ
แก้ไขชื่อ

Foreign keys
ในตาราง

Destination

(ปกติ **DB**
designer จะ

กำหนดให้
อัตโนมัติ)

Relation Editor

Relation Name:

Relation Kind: 1:1 (Non-Identifying)

Visibility: Visible

Tables:

Source: Destination:

Foreign Keys:

Source Column	Dest. Name	Comment
SSN	EMPLOYEE_SSN	

Reference Definition

Create Reference Definition

On Delete:

On Update:

2.2.1. สร้าง 1 : 1 Relationship ชื่อ manages

Relation Editor

Relation Name: manages

Relation Kind: 1:1 (Non-Identifying)

Visibility: Visible

Tables

Source: EMPLOYEE Destination: DEPARTMENT

Foreign Keys

Source Column	Dest. Name	Comment
SSN	EMPLOYEE_SSN	

Reference Definition

Create Reference Definition

On Delete: CASCADE

On Update: CASCADE

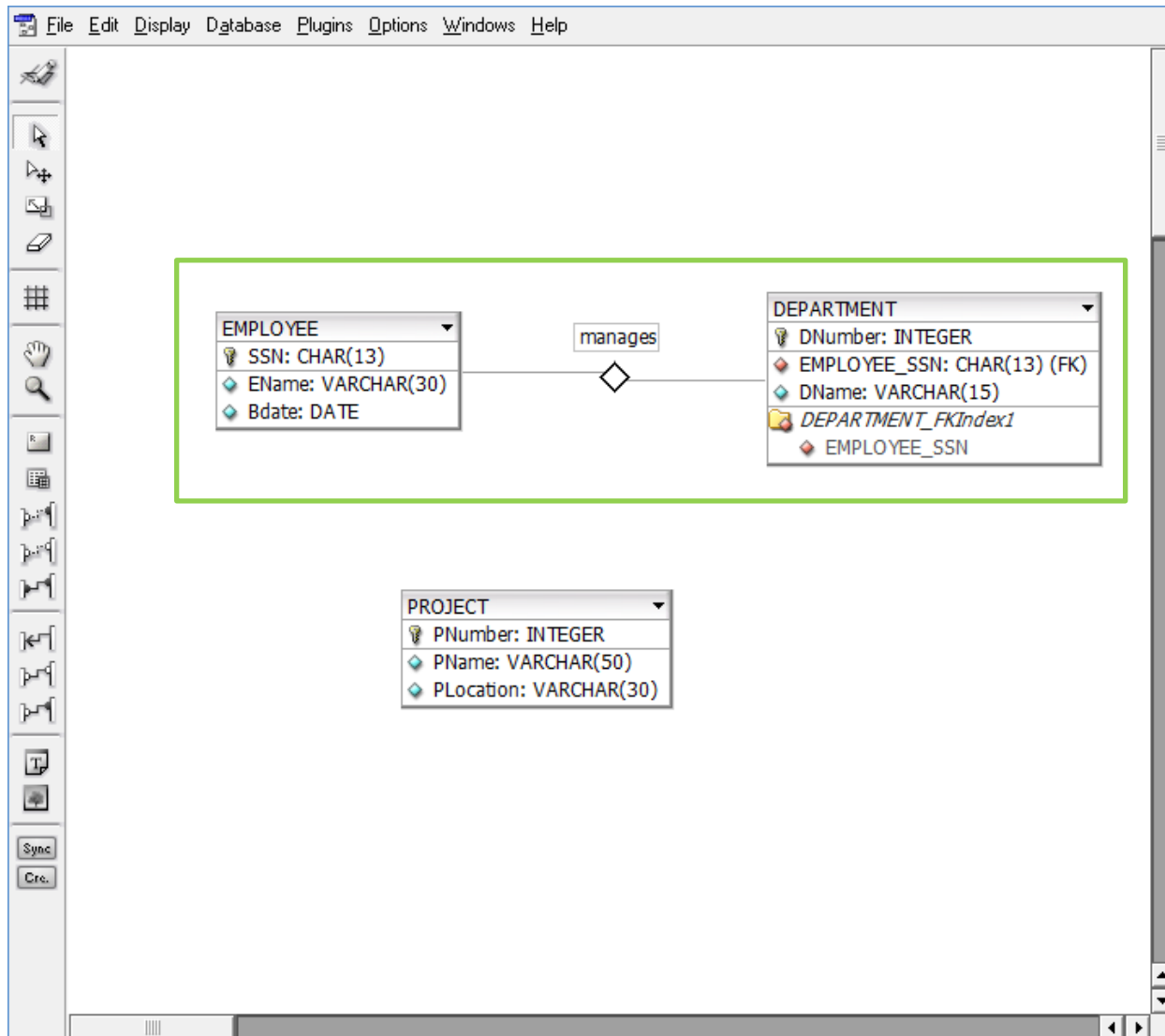
3. กำหนด **Referential integrity constraints** เมื่อมีการ **Delete** หรือ **Update** ข้อมูล อาจเลือกเป็น

- RESTRICT
- CASCADE
- SET NULL
- NO ACTION
- SET DEFAULT

4. คลิก **OK**

2.2.1. สร้าง 1 : 1 Relationship ชื่อ manages

จะได้ Manages relationship ดังรูป



2.2.2. สร้าง 1 : n Relationship ชื่อ works_for

2. คลิกที่ **Table DEPARTMENT (1)** และคลิกที่ **Table EMPLOYEE (m)**

3. ดับเบิลคลิกที่ ชื่อ **Relationship** เพื่อระบุรายละเอียด

1. เลือกเครื่องมือสร้าง **1 : n Relationship**

New 1:n Non-Identifying-Relation [R] (Click on one table, then on the other table to link these two tables.)

2.2.2 สร้าง 1 : n Relationship ชื่อ works_for

Relation Editor

Relation Name: works_for

Relation Kind: 1:n (Non-Identifying)

Visibility: Visible

Tables

Source: DEPARTMENT Destination: EMPLOYEE

Foreign Keys

Source Column	Dest. Name	Comment
DNumber	DEPARTMENT_DN	

Reference Definition

Create Reference Definition

On Delete: CASCADE

On Update: CASCADE

4. กำหนดรายละเอียดใน
Relation editor ตามปกติ
ดังนี้

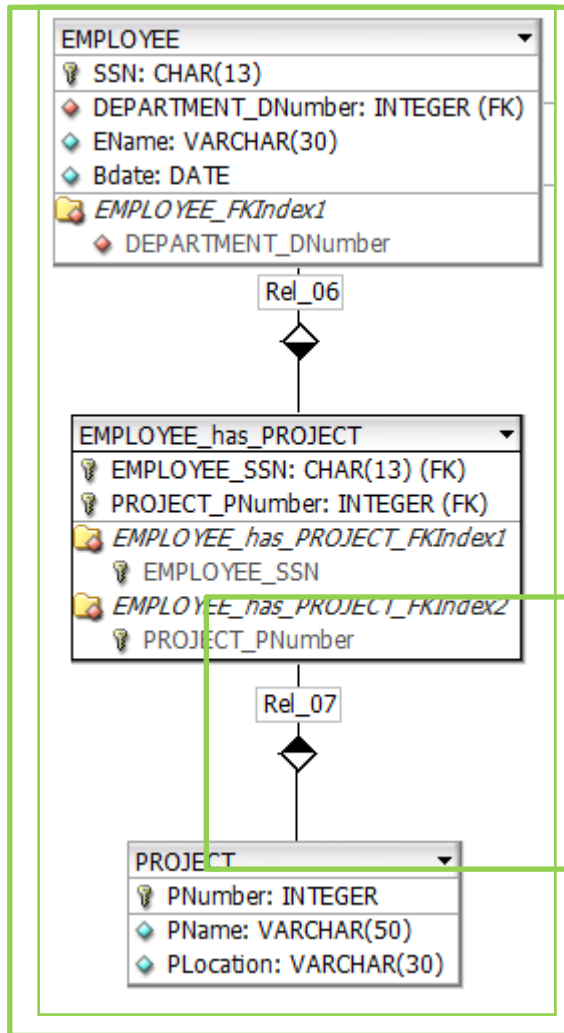
2.2.3. สร้าง m : n Relationship ชื่อ works_on



New n:m Relation [] (Click on one table, then on the other table to link these two tables by a n:m relation.)

1. เลือกเครื่องมือสร้าง **m : n Relationship**

2.2.3. สร้าง $m : n$ Relationship ชื่อ works_on



2. คลิกที่ **Table EMPLOYEE (m)** และ
คลิกที่ **Table PROJECT (n)**

3. ดับเบิลคลิกที่ ชื่อ
Relationship เพื่อระบุ
รายละเอียด

หมายเหตุ: ความสัมพันธ์แบบ $m : n$ ใน DBDesigner

จะสร้างความสัมพันธ์ให้เป็น Relation ดังนั้นจึงเกิด Relation ใหม่
ชื่อ EMPLOYEE_has_PROJECT โดยการนำ PK ทั้งสองฝั่งมาใส่
เพื่อโยงไปหาข้อมูลใน EMPLOYEE และ PROJECT

2.2.3. สร้าง m : n Relationship ชื่อ works_on

Relation Editor

Relation Name: has_worked_on Relation Kind: 1:n Visibility: Visible

Tables
Source: EMPLOYEE Destination: EMPLOYEE_has_PROJECT

Foreign Keys

Source Column	Dest. Name	Comment
SSN	EMPLOYEE_SSN	

Reference Definition

Create Reference Definition

On Delete: CASCADE

On Update: CASCADE

4. กำหนดรายละเอียดใน **Relation editor** ตามปกติ ระหว่าง

EMPLOYEE และ

EMPLOYEE_has_PROJECT

ดังรูป

โดยที่:

Source คือ **EMPLOYEE**

Destination คือ **EMPLOYEE_has_PROJECT**

2.2.3. สร้าง m : n Relationship ชื่อ works_on

Relation Editor

Relation Name: is_worked_by Relation Kind: 1:n Visibility: Visible

Tables
Source: PROJECT Destination: YEE_has_PROJECT

Source Column	Dest. Name	Comment
PNumber	PROJECT_PNumber	

Reference Definition

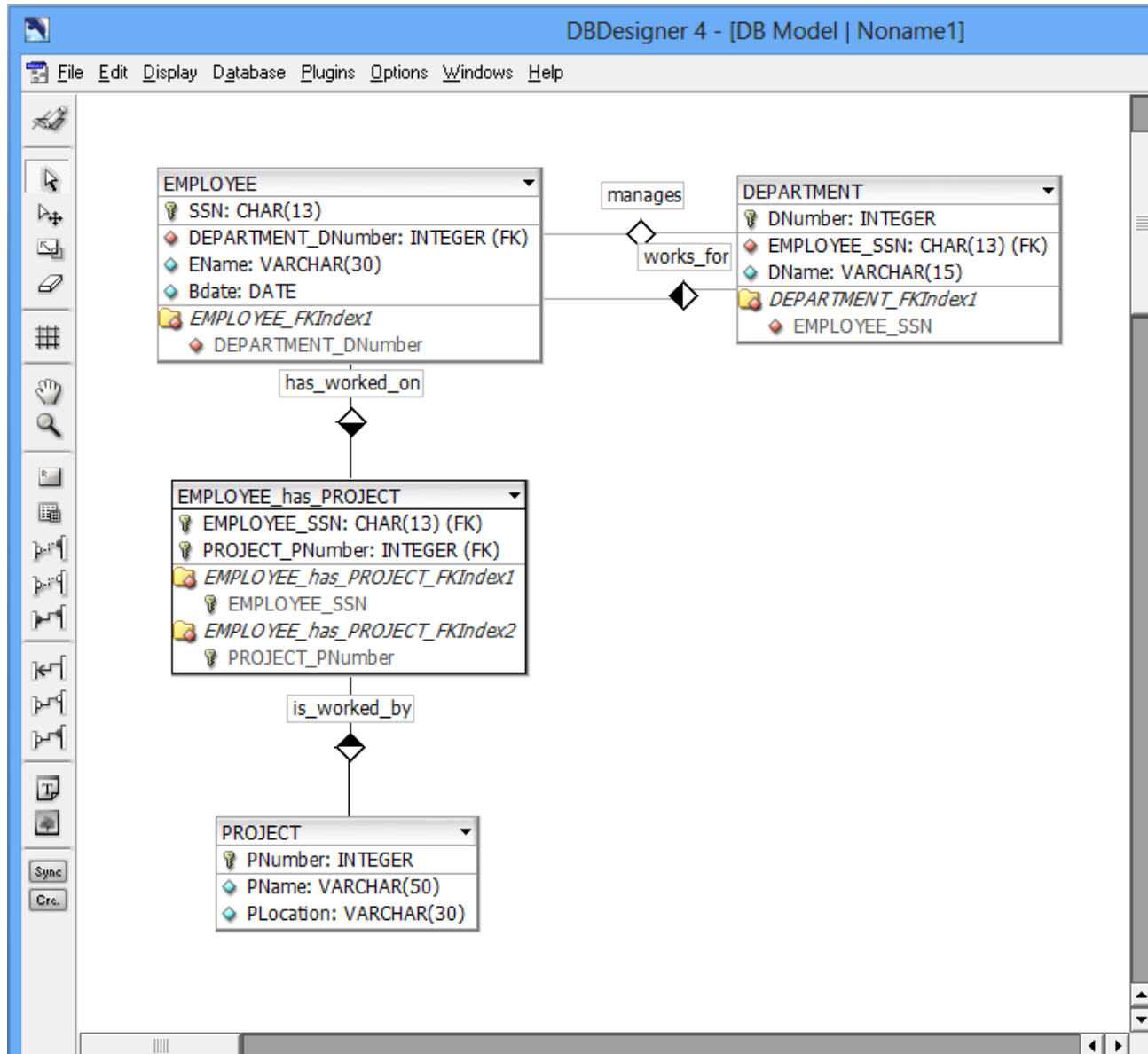
Create Reference Definition

On Delete: CASCADE

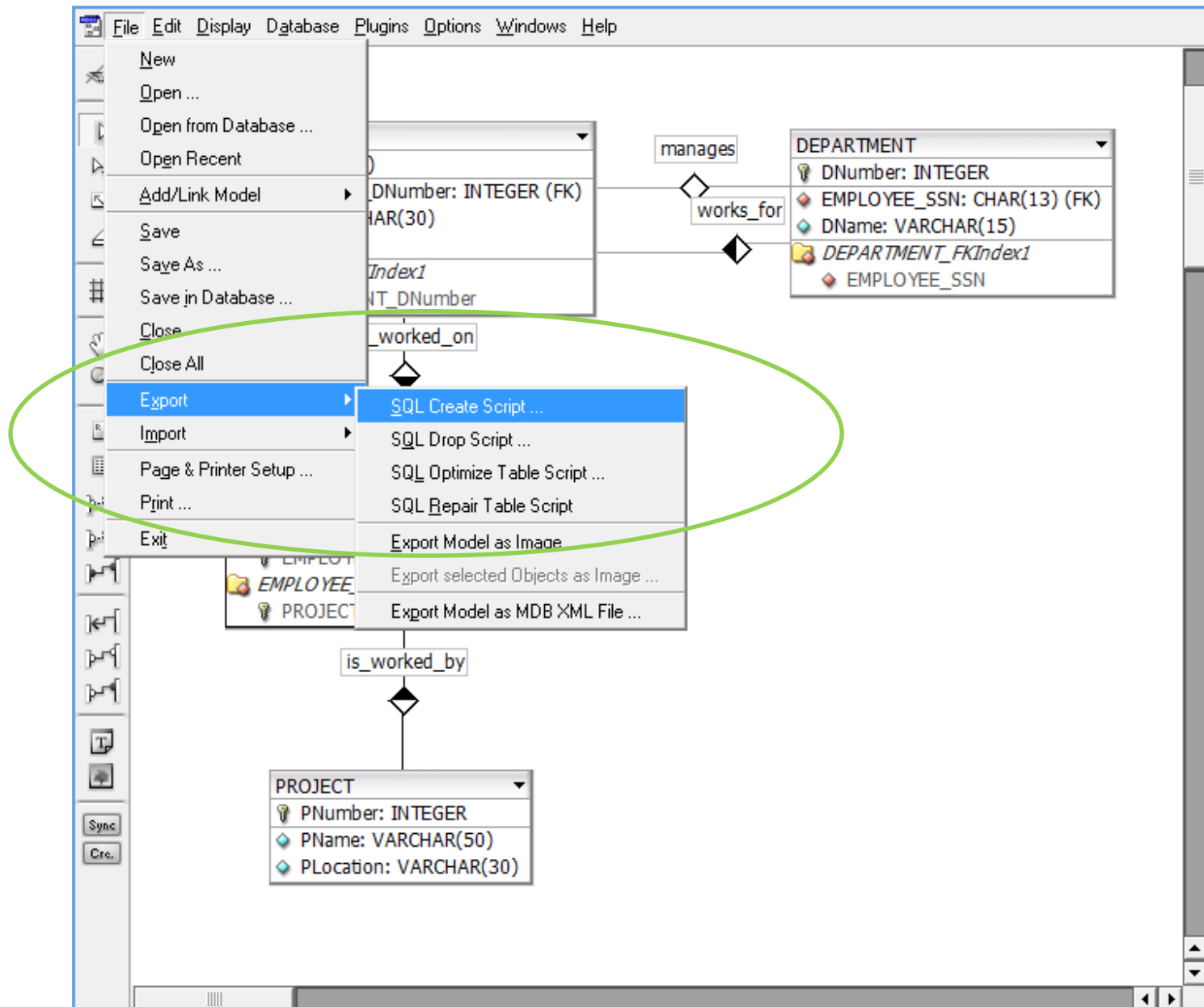
On Update: CASCADE

4. กำหนดรายละเอียดใน
Relation editor ของ
ความสัมพันธ์ ตามปกติ ระหว่าง
PROJECT และ
EMPLOYEE_has_PROJECT
ดังรูป
โดยที่:
Source คือ **PROJECT**
Destination คือ
EMPLOYEE_has_PROJECT

สรุป ER Data model ที่ได้

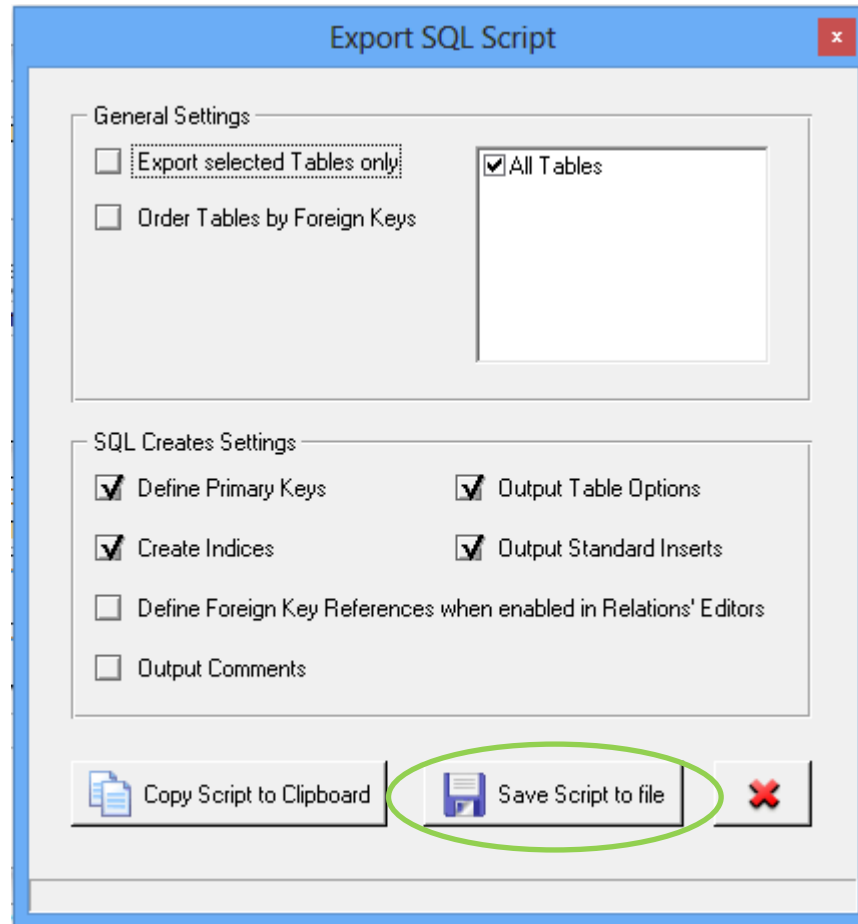


3. Export data model to the SQL script file

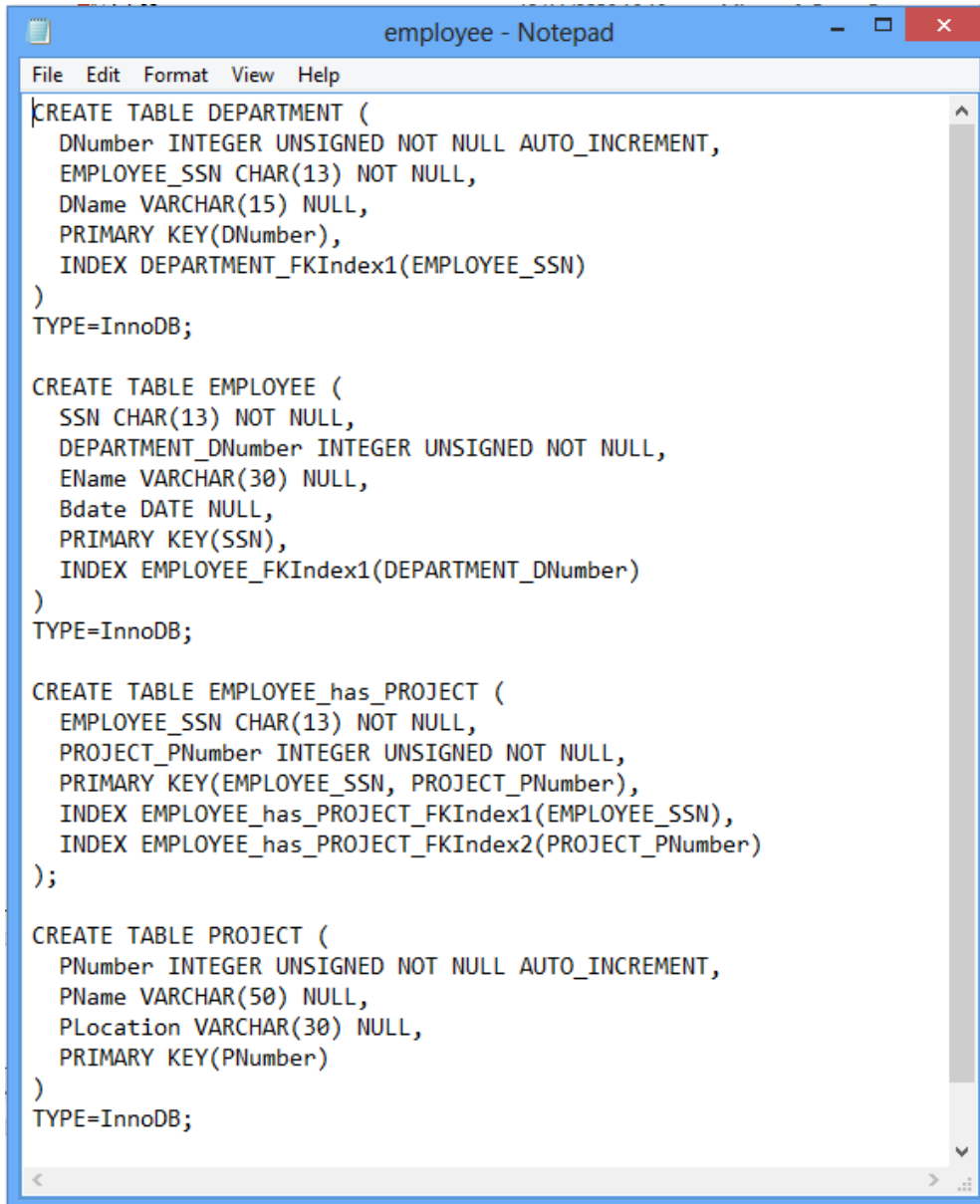


3. Export data model to the SQL script file

เลือก **Table** และ **Options** ที่ต้องการ **Export** จากนั้นเลือก **Save Script to file**



3. Export data model to the SQL script file



```
File Edit Format View Help
CREATE TABLE DEPARTMENT (
  DNumber INTEGER UNSIGNED NOT NULL AUTO_INCREMENT,
  EMPLOYEE_SSN CHAR(13) NOT NULL,
  DName VARCHAR(15) NULL,
  PRIMARY KEY(DNumber),
  INDEX DEPARTMENT_FKIndex1(EMPLOYEE_SSN)
)
TYPE=InnoDB;

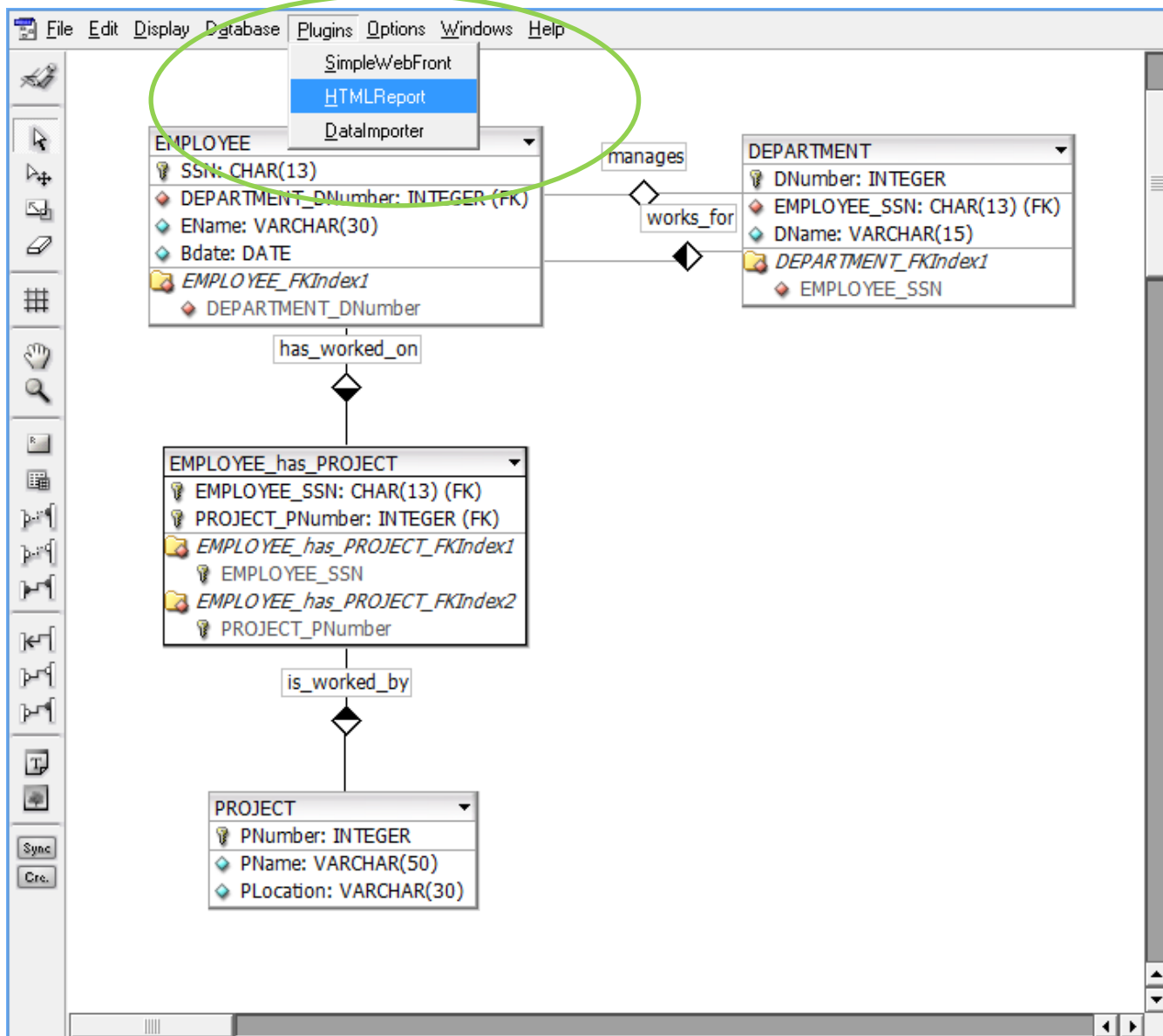
CREATE TABLE EMPLOYEE (
  SSN CHAR(13) NOT NULL,
  DEPARTMENT_DNumber INTEGER UNSIGNED NOT NULL,
  EName VARCHAR(30) NULL,
  Bdate DATE NULL,
  PRIMARY KEY(SSN),
  INDEX EMPLOYEE_FKIndex1(DEPARTMENT_DNumber)
)
TYPE=InnoDB;

CREATE TABLE EMPLOYEE_has_PROJECT (
  EMPLOYEE_SSN CHAR(13) NOT NULL,
  PROJECT_PNumber INTEGER UNSIGNED NOT NULL,
  PRIMARY KEY(EMPLOYEE_SSN, PROJECT_PNumber),
  INDEX EMPLOYEE_has_PROJECT_FKIndex1(EMPLOYEE_SSN),
  INDEX EMPLOYEE_has_PROJECT_FKIndex2(PROJECT_PNumber)
);

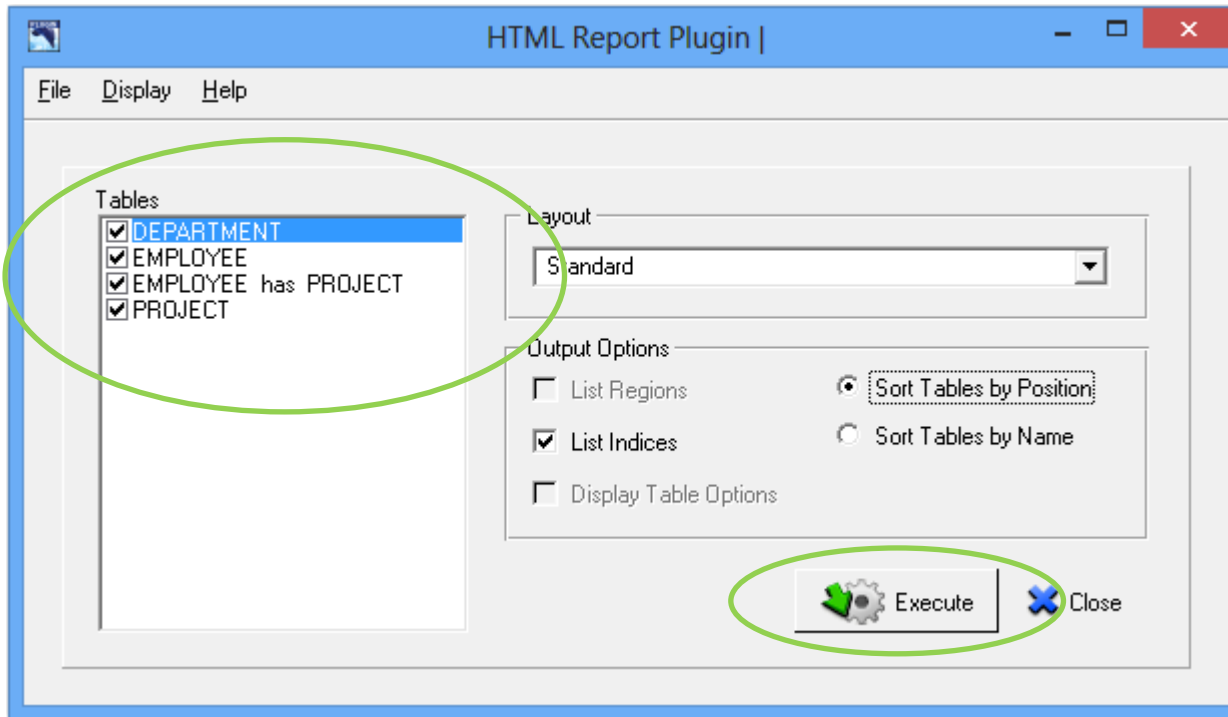
CREATE TABLE PROJECT (
  PNumber INTEGER UNSIGNED NOT NULL AUTO_INCREMENT,
  PName VARCHAR(50) NULL,
  PLocation VARCHAR(30) NULL,
  PRIMARY KEY(PNumber)
)
TYPE=InnoDB;
```

ทำการ Save file ชื่อ **employee.sql** ใน
โฟลเดอร์ที่ต้องการ จะได้ไฟล์
script ดังรูป

4. การสร้างพจนานุกรมข้อมูล (Data dictionary)

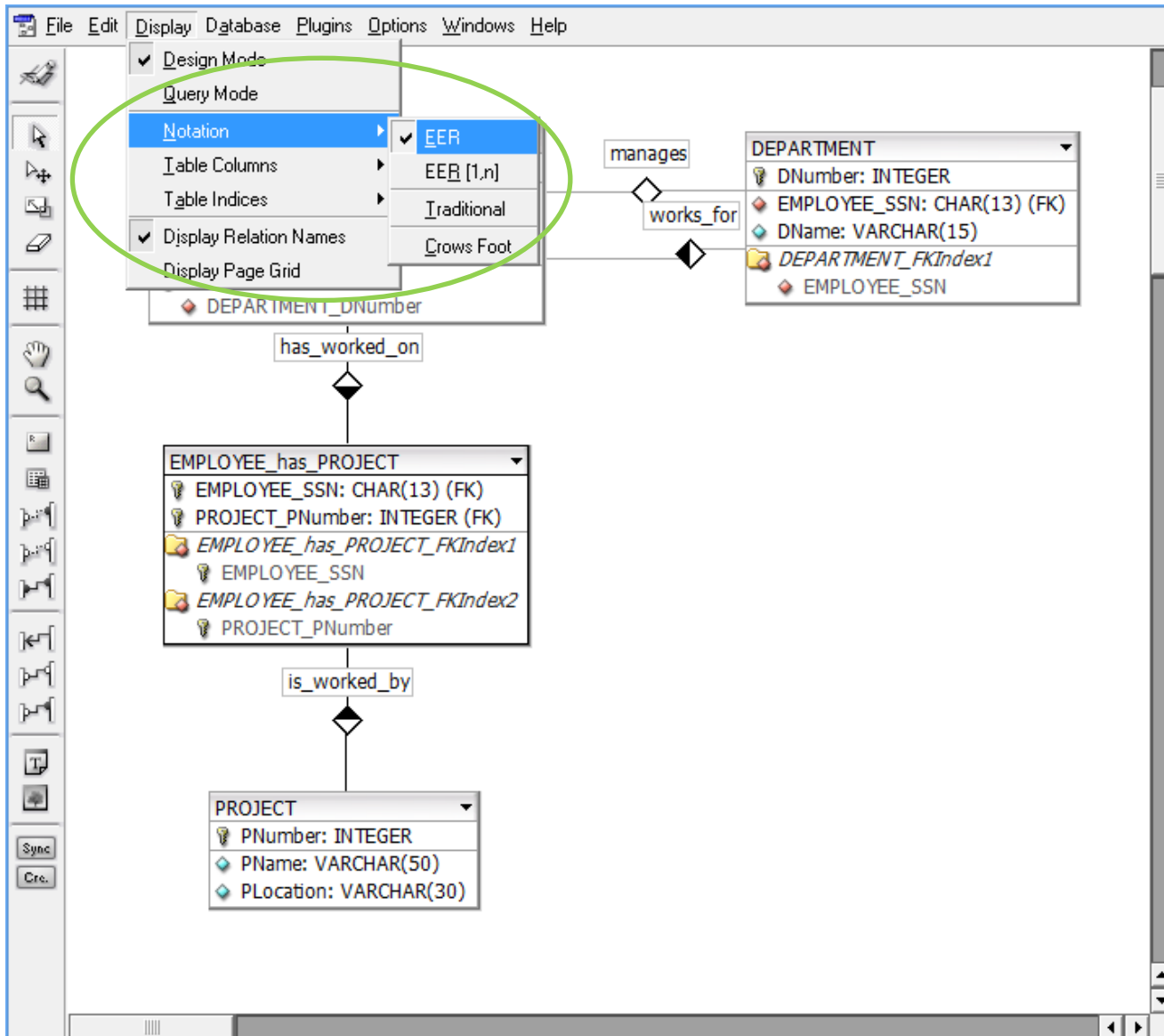


4. การสร้างพจนานุกรมข้อมูล (Data dictionary)



เลือก **Table** ที่ต้องการให้แสดงใน พจนานุกรมข้อมูล แล้วคลิก **Execute**

5. แบบฝึกหัดปฏิบัติการที่ 1



ข้อที่ 1. ให้นำส.ไปที่
เมนูดังรูป จงศึกษา
สัญลักษณ์ที่ใช้แทน
Cardinality
ratio (1:1, 1:n,
m:n) ที่มีให้ใน DB
Designer

5. แบบฝึกหัดปฏิบัติการที่ 1 (ต่อ)

ข้อที่ 2. จงสร้างผัง ER model ด้วยโปรแกรม DB Designer

- คณะวิทยาศาสตร์และเทคโนโลยีประกอบด้วยภาควิชา 10 ภาควิชา
- แต่ละภาควิชามีห้องปฏิบัติการคอมพิวเตอร์ที่มีคอมพิวเตอร์อย่างน้อย 20 เครื่องขึ้นไป สำหรับให้บริการนักศึกษา
- แต่ละห้องปฏิบัติการจะเปิดให้บริการนักศึกษาในช่วงเวลาที่ต่างกัน เช่น ห้อง **PB333** เปิดระหว่าง 9.30-12.00น. ห้อง **MB665** เปิดระหว่าง 10.00-16.00น. เป็นต้น

คณะวิทยาศาสตร์ต้องการสร้างระบบฐานข้อมูลเพื่อบันทึกข้อมูลเกี่ยวกับ

- ภาควิชา เช่น หัวหน้าภาควิชา หมายเลขโทรศัพท์ จำนวนนักศึกษา
- คอมพิวเตอร์ของภาควิชา เช่น หมายเลขประจำเครื่อง ยี่ห้อ ราคา ร้านที่จำหน่าย
- บริษัทที่จำหน่ายคอมพิวเตอร์ เพื่อติดต่อ ซ่อมแซมหรือขอใช้บริการต่างๆ เมื่อคอมพิวเตอร์มีปัญหา
- ห้องปฏิบัติการคอมพิวเตอร์ เช่น หมายเลขห้อง อาคาร ผู้รับผิดชอบดูแลห้อง เวลาเปิด-ปิด

วิธีส่งงาน แบบฝึกหัดที่ 1

ส่งงานเฉพาะข้อที่2 ดังนี้

- 1) ไฟล์รูป .jpg ของ ER model
ตั้งชื่อว่า “รหัสนักศึกษา_lab1_ER.jpg”
- 2) ไฟล์ script.sql ของ ER model
ตั้งชื่อว่า “รหัสนักศึกษา_lab1_ER.sql”

กำหนดส่งงาน: **ท้ายชั่วโมงปฏิบัติการ**