



EPL342 –Databases

Lab 2

**ER Modeling (Entities) in DDS Lite &
Conceptual Modeling in SQL Server 2008**

Panayiotis Andreou

<http://www.cs.ucy.ac.cy/courses/EPL342>



Before We Begin

- Start the DDS Lite
 - Start → All Programs → Chilli Source → DDS-Lite
- Start the SQL Server Management Studio
 - Start → All Programs → Microsoft SQL Server → SQL Server Management Studio

Server: APOLLO.IN.CS.UCY.AC.CY

Authentication: SQL Server Authentication

Username: <check your email>

Password: <check your email>

Lab Introduction



COMPANY Database

- During your [lecture 4](#), you have identified 4 entities consisting of the COMPANY db:
 - DEPARTMENT
 - PROJECT
 - EMPLOYEE
 - DEPENDENT
- Our job is to design the entities/tables based on the requirements



DDS Lite – Create new Project

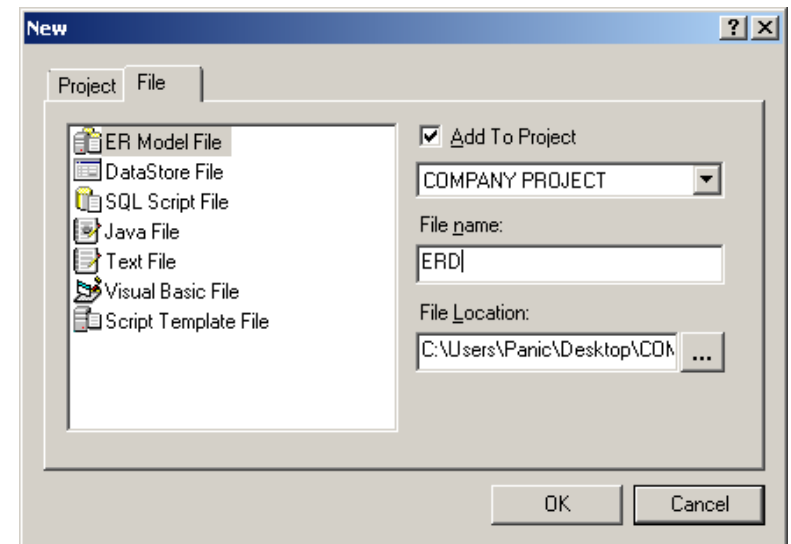
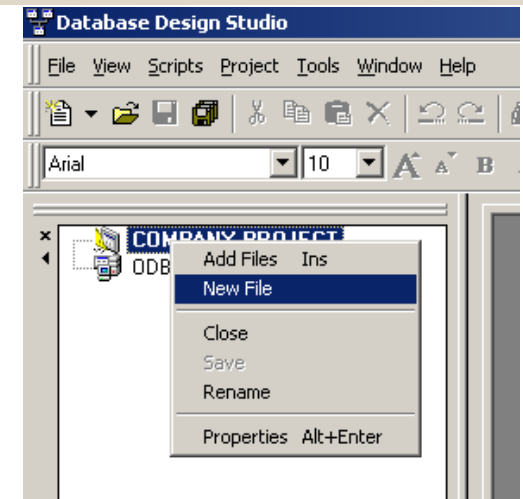
- Create a new project in DDS (File→New)
 - Project Name: COMPANY PROJECT
 - Location: <your desired location>
 - Project Title: COMPANY PROJECT
 - Author: <your name>

A screenshot of the 'New' dialog box in the DDS Lite software. The dialog box has a title bar with 'New' and standard window controls. It contains four input fields: 'Project Name' with the text 'COMPANY', 'Location' with the path 'C:\Program Files\DDS_Lite\Projects\COMPANY' and a browse button (...), 'Project Title' with the text 'DDS Project', and 'Author' with the text 'Panayiotis Andreou'. At the bottom right, there are 'OK' and 'Cancel' buttons.



DDS Lite – Create new File

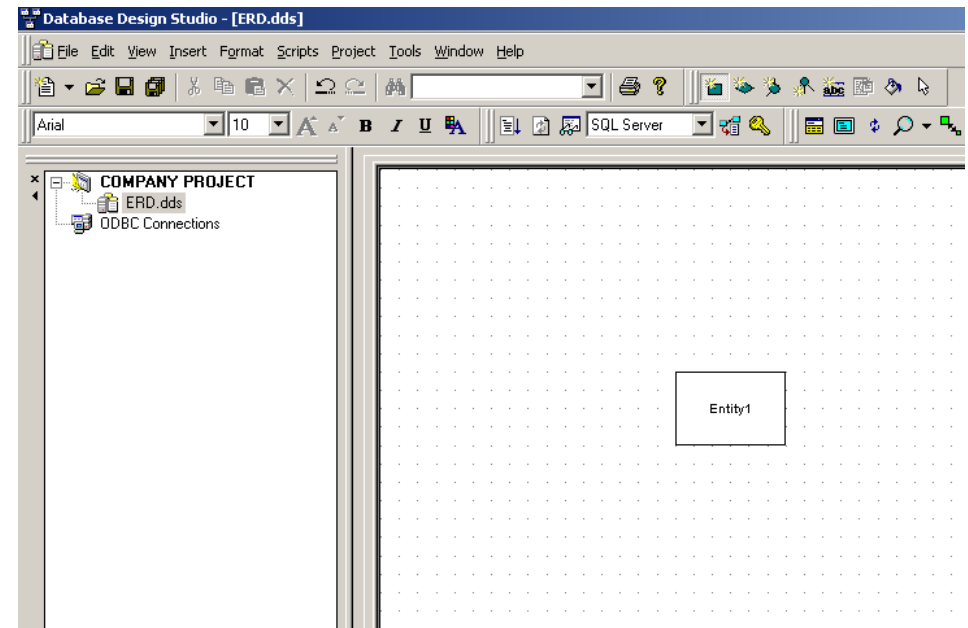
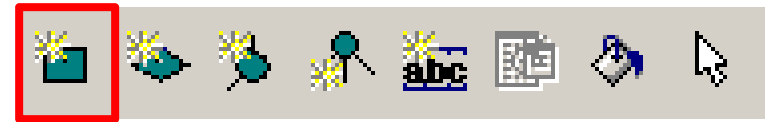
- Create a new file on the COMPANY PROJECT (Right-click → New file)
- Choose the ER Model file
- Type ERD as the Filename





DDS Lite – Create new Entity

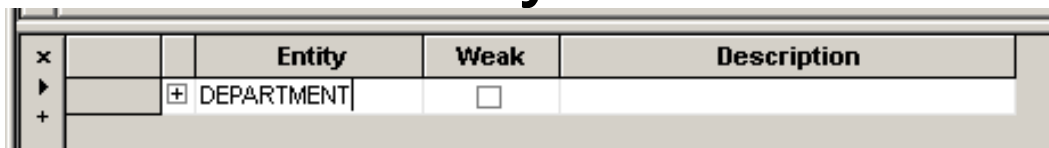
- To create a new entity go to Insert → Entity or
- Next, click on the main panel of DDS
- A new entity (Entity 1) will be created





DDS Lite – Entity Properties

- To change the properties of an entity right click on the entity and select properties
- Rename Entity1 to DEPARTMENT



- Click on the icon and enter the following attributes:

DEPARTMENT												
	Key	Attribute	Datatype	Precision	Scale	Modifier	Nullable	Serial	Default	Check	Unique	Comment
<input type="checkbox"/>	<input checked="" type="checkbox"/>	DNumber	INTEGER				<input type="checkbox"/>	<input type="checkbox"/>			<input checked="" type="checkbox"/>	
	<input type="checkbox"/>	DName	VARCHAR	50			<input type="checkbox"/>				<input checked="" type="checkbox"/>	
	<input type="checkbox"/>	Location	VARCHAR	50			<input type="checkbox"/>				<input type="checkbox"/>	
*	<input type="checkbox"/>						<input type="checkbox"/>	<input type="checkbox"/>			<input type="checkbox"/>	

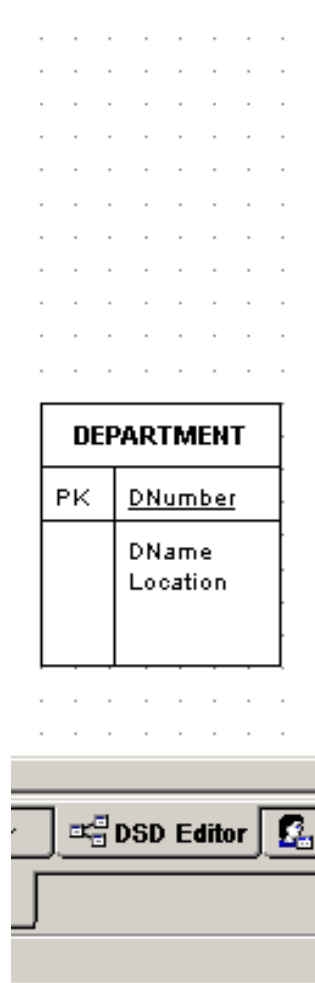


DDS Lite – Different Views

ERD Editor



DSD Editor



Schema (SQL)

```
USE db_name
GO

IF DB_NAME() = 'db_name'
    RAISERROR('db_name' DATABASE CONTEXT NOW :
ELSE
    RAISERROR('ERROR IN BATCH FILE, 'USE db_name

GO
EXECUTE SP_DBOPTION 'db_name' , 'TRUNC. LOG ON CHI
GO

--
-- Create Table      : 'DEPARTMENT'
-- DNumber           :
-- DName             :
-- Location          :
--
CREATE TABLE DEPARTMENT (
    DNumber          INT NOT NULL UNIQUE,
    DName            VARCHAR(50) NOT NULL UNIQUE,
    Location         VARCHAR(50) NOT NULL,
CONSTRAINT pk_DEPARTMENT PRIMARY KEY CLUSTERED (I
GO

--
-- Permissions for: 'public'
--
GRANT ALL ON DEPARTMENT TO public
GO
```

The Schema (SQL) window displays the SQL code for creating the "DEPARTMENT" table. The window title bar shows "Schema". The taskbar shows "Schema", "Information", and "Datatypes".



DDS Lite – Practice (10 minutes)

- Create Entities and Attributes in DDS Lite for the following:

EMPLOYEE

- Ssn (key, unique)
- Bdate
- Fname
- Minit
- Lname
- Address
- Salary
- Sex

PROJECT

- **Number** (key, unique)
- **Name**
- Location
- Department

DEPENDENT

- Employee
- **Name**
- Relationship
- Bdate
- Sex

- Choose the appropriate data types



Lab 2

Conceptual Modeling in SQL Server 2008



Creating Tables

First, navigate to your database and right-click on Tables → New Table

The designer provides you with 3 columns:

Column Name	Data Type	Allow Nulls
		<input type="checkbox"/>

Column name: the name of the column (e.g., Name, Birth Date, Salary)

Data type: the data type for the column (e.g., int, varchar(30), bit)

Allow nulls: if checked then you must supply a value for each row (nulls are not allowed)



DEPARTMENT

Specification I - Each department has the following fields:

- Name
- Number
- Manager
- Start date of the department manager
- **Multiple locations**

Creating Table DEPARTMENT



You must select the appropriate data type for each column

- **number:** is an integer (e.g., tinyint, smallint, int, bigint)
- **name:** is a string (e.g., char, varchar)
- **Manager:** is the name of an employee (i.e., a string)
- **Manager start date:** is a date (e.g., datetime, smalldatetime)



Creating Table DEPARTMENT

- Create table department using the following specifications

Column Name	Data Type	Allow Nulls
number	int	No
name	nvarchar(50)	No
Manager	nvarchar(50)	No
Manager start date	Smalldatetime	No

- Save the table with the name DEPARTMENT



Column Names

- Column names can only contain valid characters (i.e., letters, digits and underscores)
- When you saved the DEPARTMENT table note that **<Manager start date>** is now saved as **<[Manager start date]>**. This is because white spaces are considered invalid characters.



Column Names Limitations

To ensure maximum database compatibility

- **Avoid using special characters**

If you use these characters < [,], ', " > then it is more difficult not only to reference that object but also to read code that contains the name of that object

- **Avoid using reserved keywords**

SQL Server uses [reserved keywords](#) for manipulating and accessing databases.

→ When you use special characters or reserved keywords the column name is saved using brackets



Column Names/Identifier Rules

1. The first character must be one of the following:
 - A letter as defined by the Unicode Standard 3.2. The Unicode definition of letters includes Latin characters from a through z, from A through Z, and also letter characters from other languages.
 - The underscore (`_`), at sign (`@`), or number sign (`#`).

Certain symbols at the beginning of an identifier have special meaning in SQL Server. A regular identifier that starts with the at sign always denotes a local variable or parameter and cannot be used as the name of any other type of object. An identifier that starts with a number sign denotes a temporary table or procedure. An identifier that starts with double number signs (`##`) denotes a global temporary object. Although the number sign or double number sign characters can be used to begin the names of other types of objects, we do not recommend this practice. Some Transact-SQL functions have names that start with double at signs (`@@`).
2. Subsequent characters can include the following:
 - Letters as defined in the Unicode Standard 3.2.
 - Decimal numbers from either Basic Latin or other national scripts.
 - The at sign, dollar sign (`$`), number sign, or underscore.
3. The identifier must not be a Transact-SQL reserved word. SQL Server reserves both the uppercase and lowercase versions of reserved words.
4. Embedded spaces or special characters are not allowed.
5. Supplementary characters are not allowed

[More info here](#)



Column Properties

Table: Department
Column: number

Column Properties	
(General)	
(Name)	number
Allow Nulls	No
Data Type	int
Default Value or Binding	
Table Designer	
Collation	<database default>
Computed Column Specification	
Condensed Data Type	int
Description	
Deterministic	Yes
DTS-published	No
Full-text Specification	
Has Non-SQL Server Subscriber	No
Identity Specification	
Indexable	Yes
Merge-published	No
Not For Replication	No
Replicated	No
RowGuid	No
Size	4
(General)	



COMPANY Database

- Create the COMPANY database using the following specifications

EMPLOYEE

Column Name	Data Type	Allow Nulls
SSN	int	<input type="checkbox"/>
Bdate	smalldatetime	<input type="checkbox"/>
Fname	nvarchar(20)	<input type="checkbox"/>
Minit	nvarchar(1)	<input type="checkbox"/>
Lname	nvarchar(30)	<input type="checkbox"/>
Address	nvarchar(100)	<input type="checkbox"/>
Salary	smallmoney	<input type="checkbox"/>
Sex	bit	<input type="checkbox"/>
Department	nvarchar(50)	<input type="checkbox"/>
Supervisor	int	<input type="checkbox"/>

DEPENDENT

Column Name	Data Type	Allow Nulls
Relationship	nvarchar(30)	<input type="checkbox"/>
Birth_date	smalldatetime	<input type="checkbox"/>
Sex	bit	<input type="checkbox"/>
Employee	int	<input type="checkbox"/>
Dependent_name	nvarchar(50)	<input type="checkbox"/>

PROJECT

Column Name	Data Type	Allow Nulls
number	int	<input type="checkbox"/>
name	nvarchar(50)	<input type="checkbox"/>
location	nvarchar(50)	<input type="checkbox"/>
controlling_department	nvarchar(50)	<input type="checkbox"/>



Primary Keys

A **Primary key** is a candidate key to uniquely identify each row in a table

Candidate keys:

DEPARTMENT: name, number


PROJECT: name, number

EMPLOYEE: ssn

DEPENDENT: ???



Creating Primary Keys


- To create a Primary Key, open the table design and select a column.
- Use the  icon to assign the column as the primary key

Set the following Primary Keys

DEPARTMENT: number

PROJECT: number

EMPLOYEE: ssn

DEPARTMENT	
	number
	name
	Manager
	Manager_start_date



Creating Identities

Identities are columns with unique values produced automatically from SQL Server.

You can set the following properties:

- **Is Identity** Indicates whether or not this column is an identity column.
- **Identity Seed:** The value that will be assigned to the first row in the table.
- **Identity Increment:** This value is the increment that will be added to the **Identity Seed** for each subsequent row.





Database Diagram

- Right-click on Database Diagrams and select **New Database Diagram**
- Use the add button to add all tables to the diagram
- Right-click on a table to see Table View options (e.g., Standard, Keys, Custom, etc)
- Save the diagram as DD
- **In the future we are going to add relationships to the database diagram**

COMPANY database diagram



DEPARTMENT	
 number	
name	
Manager	
Manager_start_date	

EMPLOYEE	
 SSN	
Bdate	
Fname	
Minit	
Lname	
Address	
Salary	
Sex	
Department	
Supervisor	

DEPENDENT	
Relationship	
Birth_date	
Sex	
Employee	
Dependent_name	


PROJECT	
 number	
name	
location	
controlling_department	



Table Views

Column Names

DEPARTMENT	
	number
	name
	Manager
	Manager_start_date

Standard

DEPARTMENT			
	Column Name	Data Type	Allow Nulls
	number	int	<input type="checkbox"/>
	name	nvarchar(50)	<input type="checkbox"/>
	Manager	nvarchar(50)	<input type="checkbox"/>
	Manager_start_date	smalldatetime	<input type="checkbox"/>

Keys

DEPARTMENT	
	number

Custom

DEPARTMENT						
	Column Name	Nullable	Data Type	Length	Default Value	Description
	number	No	int	4		
	name	No	nvarchar(50)	50		
	Manager	No	nvarchar(50)	50		
	Manager_st...	No	smalldatetime	4		