LAB 6: Code Generation with Visual Paradigm for UML and JDBC Integration

OBJECTIVES

- To understand the steps involved in Generating codes from UML Diagrams in Visual Paradigm for UML.
- Exposure to JDBC integration and interaction of SQL database with Java programs.

INTRODUCTION

Development and deployment (both programming codes and database) presents the next step in Software development life cycle, especially after Requirement Modeling and Analysis has been carried out.

UML has two diagrams that are used for behavior specification: the activity diagram and the state diagram. These two diagrams (more one than the other, depending on the nature of the subsystem being modeled).

The class diagram (for modeling the structural aspects of the object model) provides a framework that supports the design of complex applications in a way that is fully complete (and thus allowing 100% code generation) while still implementation independent (which means that the programmer will write the rest of the implementation codes).
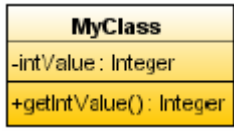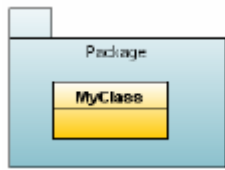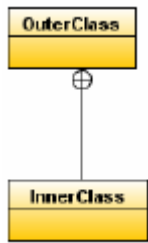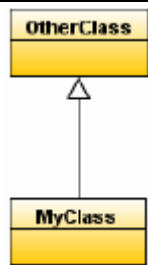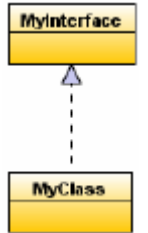
All the other diagrams (use case, sequence, collaboration) are interesting for gathering requirements, but are useless in modeling a solution that can be automatically transformed into a running application, and thus we will ignore them here.

For this lab, we will cover on how to implement codes from Class Diagram.

Model Representation of Code from Class Diagram

Type	Model	Code
Class		public class MyClass{ }
Abstract Class		public abstract class MyClass { }

Attribute		public class MyClass
		{
		   private int intValue;
		}

| Operation		public class MyClass
		{
		   private int intValue;
		   private setValue() {}
		}

Package		package Package; public class MyClass {}

Inner Class		public class MyClass
		{
		   public class InnerClass {}
		}

Generalization		public class MyClass extends OtherClass {}
Realization		Public class MyClass implements MyInterface {}

Association		Public class MyClass
		{
		   OtherClass otherClass;
		}

CODE GENERATION FROM UML DIAGRAMS

Other than allowing code generation from UML models, VP UML provides a feature to automatically create a JAVA project in Eclipse / Netbeans IDE from UML diagrams. Unfortunately, such feature is only available in the Professional Edition and above (UKM labs are installed with Standard licenses VP UML).

What we can use instead is the "Instant Code Generation" feature in VP UML, which automatically generates skeleton java codes (Feature available only from Standard Edition onwards, not applicable to Community Edition). From these skeleton codes, we will use them to make a JAVA project in our preferred IDE, in this case the Eclipse IDE.

At the same time, VP UML also provides database generation from UML models (for Professional edition and above only), so to develop our database, we will build them manually.

Now, consider a Class Diagram for "Customer Order System" below. Follow the steps provided to generate java codes from it.
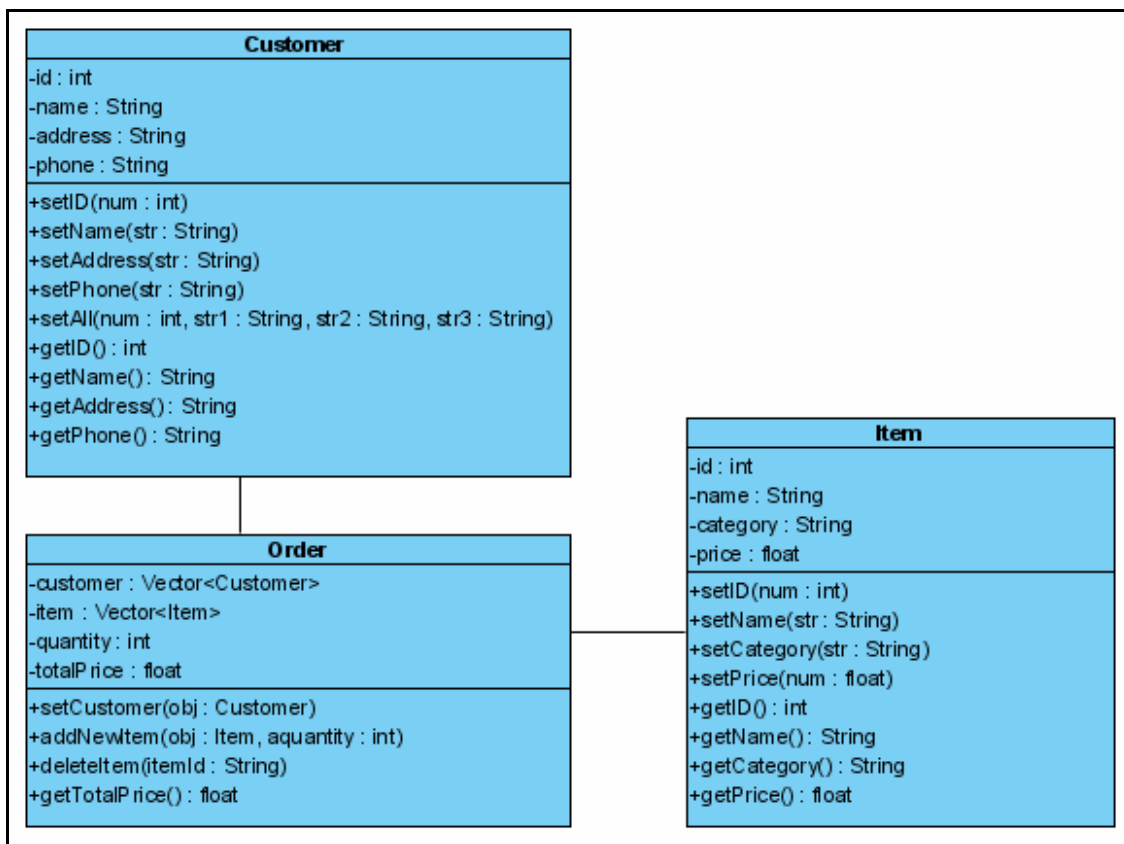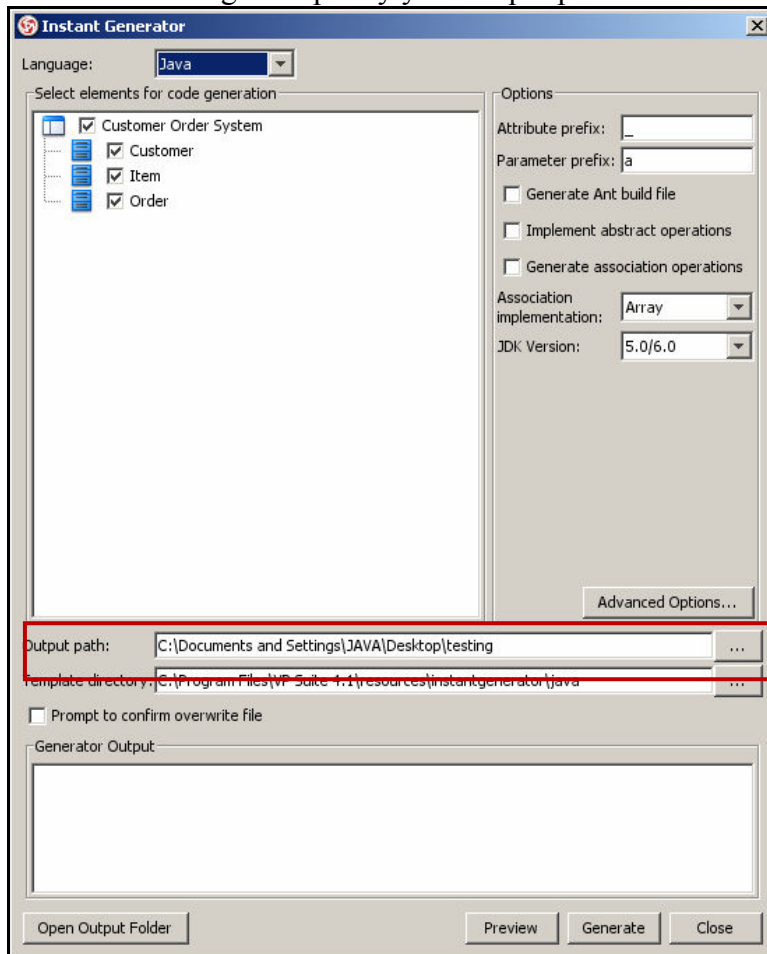

Class Diagram for Customer Order

- Under Tools > Instant Generator > Choose "Java". Refer below diagram:
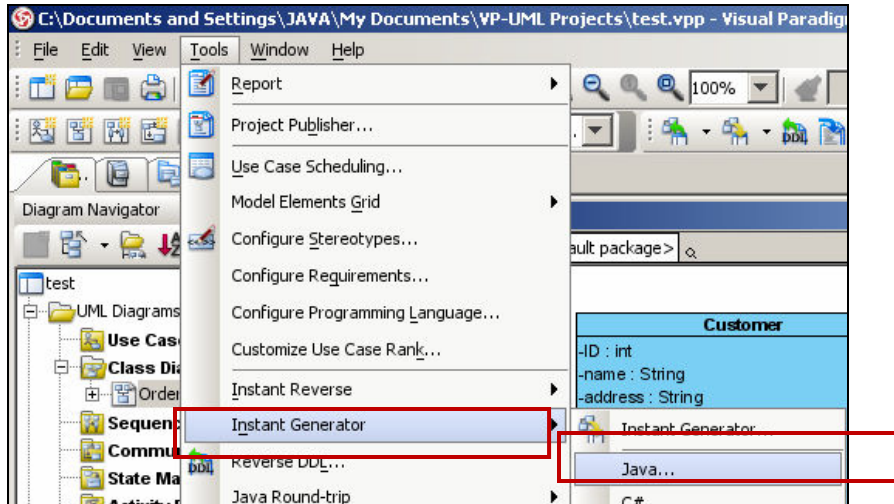
- Make sure all your settings in the Instant Generator window follows the diagram below. Don't forget to specify your output path:

JAVA CONNECTIVITY WITH SQL DATABASE USING JDBC

There are lots of SQL servers available for connection with JAVA such as MySQL, PostgreSQL or SQLite. In this lab however, we will cover on SQL Server 2005 with the Eclipse IDE.

A Java project has been provided for this lab called DBJtds (source codes), which is the code implementation of the class diagram Customer Order earlier. The codes will demonstrate how to connect JAVA to SQL Server 2005 and do SQL operations.

Before running DBJtds, a few settings on the SQL side are required:
- Database "CustomerOrder" must have already been created in SQL Server 2005.
- Table "Customer" must be created under Database "CustomerOrder".
- Account student with username "student" and password "mypassword" must be available / exist for SQL Server 2005.
- Account student must be granted access and control to database "CustomerOrder".
- Identify the port number the SQL Server is listening to.
- SQL Server must enable both Window Authentication / SQL Authentication to access its server.

Furthermore, we need a JDBC driver to enable our Java application to interact with the SQL database. Using SQL Server 2005, one driver available for free is Jtds JDBC driver. We won't go into much detail on definitions of JDBC as you're expected to have covered this (To download jtds jdbc driver, refer to Appendix A).

Environment Settings of SQL Server 2005

- Click Start > All Programs > Microsoft SQL Server 2005 > Configuration Tools > SQL Server Surface Area Configuration, as shown below:
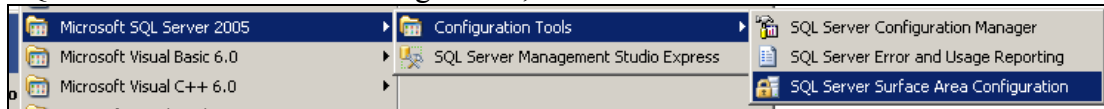
- A dialog box will popup. Choose "Surface Area Configuration for Services and Connections".
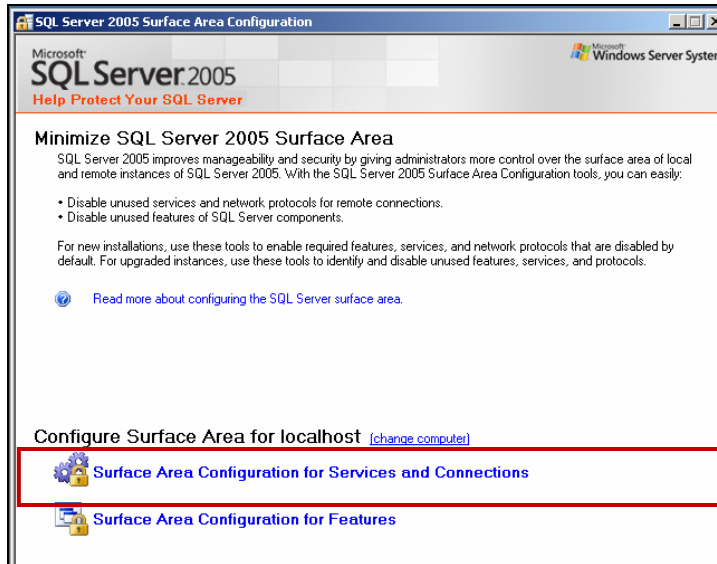
- Under the Main Menu, select Service Tab. Make sure the service is started, and not stopped. If it is stopped, then just click the "Start" button. Refer to figure below.
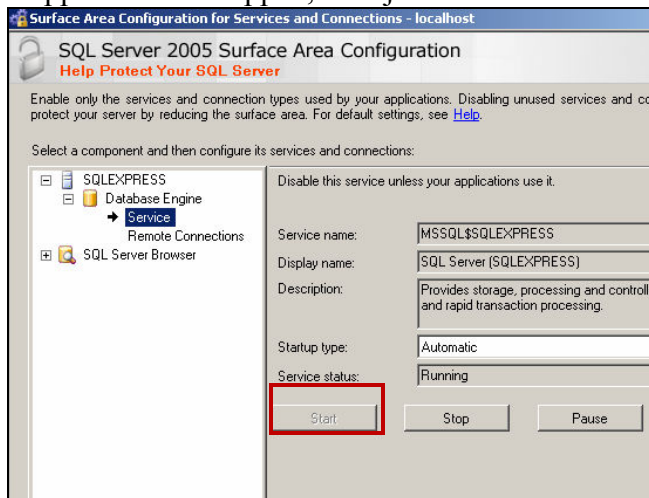

- Also make sure under "Remote Connections", choose "Local and Remote Connections" and also highlight the "Using both TCP/IP and named pipes".
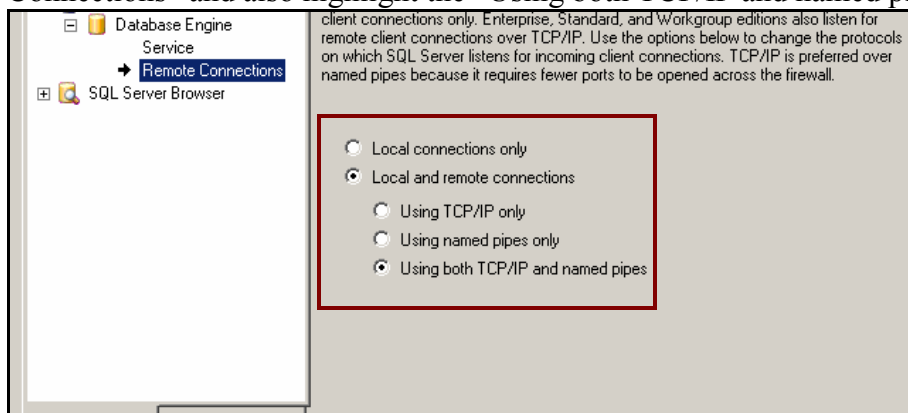
Setting the Port Number Listened by SQL Server 2005

SQL Server allocates dynamically the port number it listens to. Sometimes it's quite difficult to track the port number used by SQL Server at any one time. To solve this, we can manually set the port number used instead.

- Open SQL Server Configuration Manager(SSCM)
    Start > All Programs > Microsoft SQL Server 2005 > Configuration Tools

- Highlight "Protocols for SQLEXPRESS". Right click "TCP/IP" and choose "Properties".
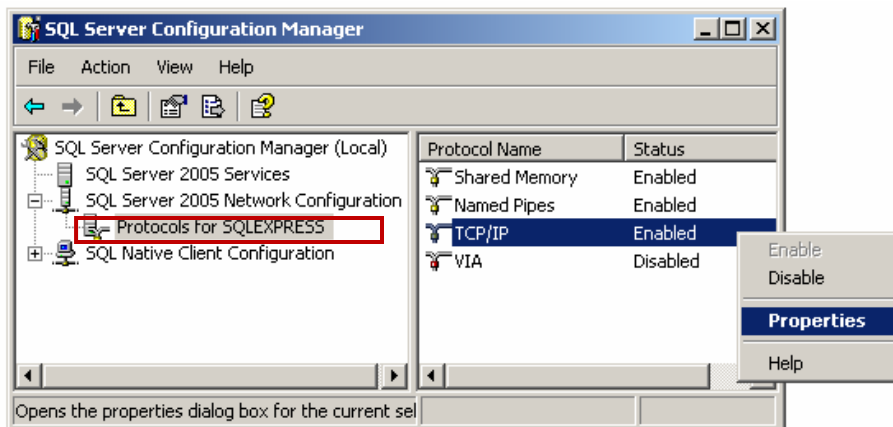
- In the Properties dialog box, set the port number to an available, free port. Port 1433 is strongly recommended.
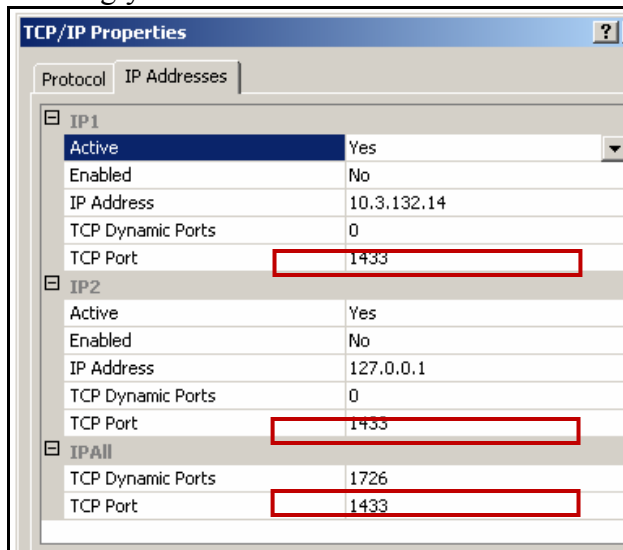
- Remember to restart the SQLServer to apply the changes.
    Under SSCM, highlight SQL Server 2005 Services > Right click SQL Server (SQLEXPRESS).
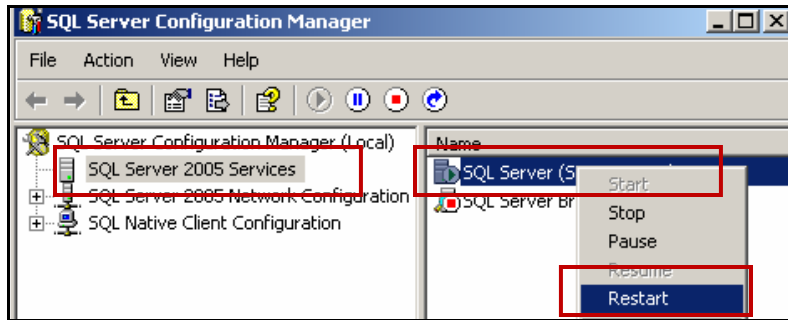
Executing sqlcmd (A Command Line Tool for Sql Server 2005)

To manage our SQL actitivites in SQL Server 2005, a command line utility called "sqlcmd" is available (Alternatively, we can use a GUI based SQL Manager called SSMSEE to achieve the same thing). Let's look at how to operate with sqlcmd first.
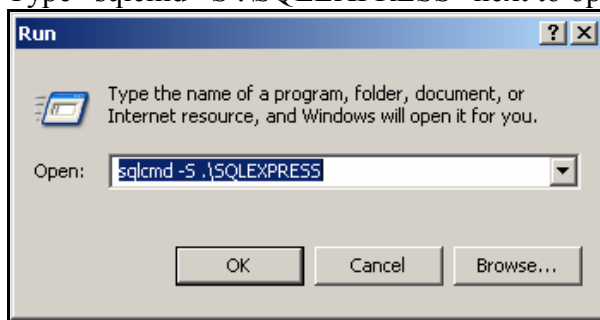
- Click Start > Run

- Type "sqlcmd –S .\SQLEXPRESS" next to open.

- You should get the following display below. You can type all sql commands here.
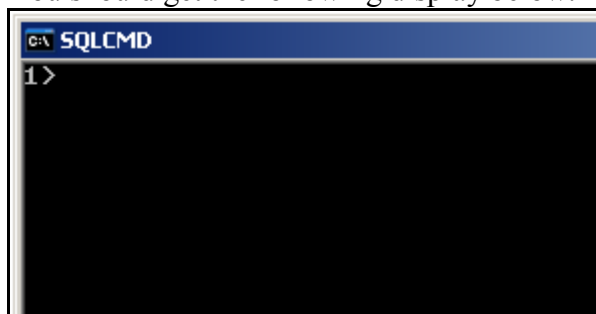
Common SQL Command line operations in sqlcmd

Below outlined are several common command lists to manage sql operations:

Command	Description / Usage
> sp_databases	List all databases in the server.

> use DATABASENAME	Make DATABASENAME the active database.
> sp_help TABLENAME	List all fields in table TABLENAME.
> drop database DATABASENAME	Delete the database DATABASENAME.
> Select * from information_schema.tables	List all tables in the active database.
> drop table TABLENAME	Delete the table TABLENAME.
> select * from TABLENAME	List all data of TABLENAME.

Note: Remember, all operations will not be executed unless they are ended by the word "go"

How to create a new database
> create database DATABASENAME
> go

How to create a new table in a database
> use DATABASENAME
> create table TABLENAME
> (att1 int PRIMARY KEY,
>   att2 varchar(50) NOT NULL)
> go

How to create a new table in database with auto increment primary key
> use DATABASENAME
> create table TABLENAME
> (att1 int PRIMARY KEY IDENTITY(1,1),
> att2 varchar(50) NOT NULL)
> go

How to modify column in TABLENAME
> alter table TABLENAME
> modify ATTRIBUTENAME varchar(50)
> go

How to drop column in TABLENAME
> alter table TABLENAME
> drop column ATTRIBUTENAME
> go

How to insert value into table TABLENAME
> insert into TABLENAME VALUES ('value1', 'value2');
> go

How to delete data from table TABLENAME
> drop from TABLENAME
> where COLUMNNAME = COLUMNVALUE

> go

How to update / change value of data in TABLENAME
> update TABLENAME
> set att1 = NEW_VALUE where att2 = ATT2_VALUE_OF_THAT_ROW
> go

<<In-Class Exercise on SQL Management with SQL Server 2005>>

To continue with our implementation of the class diagrams, we will create 2 tables:
- Customer
- Item

First, create a new database "CustomerOrder" in sqlcmd (if it's not yet created):
> create database CustomerOrder
> go

To create table Customer, type the following commands below in sqlcmd:
> create table Customer
> (
> id int PRIMARY KEY,
> name varchar(50) NOT NULL,
> address varchar(50) NOT NULL,
> phone varchar (50) NOT NULL
>)
> go

Now, fill in the data for both table Customer and Item as shown below (You don't have to exactly follow the elements listed below):

TABLE Customer

Id	Name	Address	Phone
1001	Aladdin	Bangi	03-8080 8080
1002	Archie	Bangi	03-8050 8050
1003	Johnny Quest	Kajang	03-7070 7070

TABLE Item

| Id	Name	Category	Price
10001	Cadburry 250g Almond	Chocolate	2.50
10002	Ferrero Roucher 250g	Chocolate	4.00
10005	Durian Pack 500g	Fruit	10.00

**SQL Database and Account Management with SQL Server Management Studio
Express Edition (SSMSEE) tool**

To make it easier to do management activities in SQL server, you can use SSMSEE for
2005. If you don't have it installed, follow the procedures below:

▪ Install the file SQLServer2005_SSMSEE.msi provided for this lab.

▪ Launch the SQL Server Management software SSMSEE.

▪ When first time connecting, choose authentication "Windows Authentication".
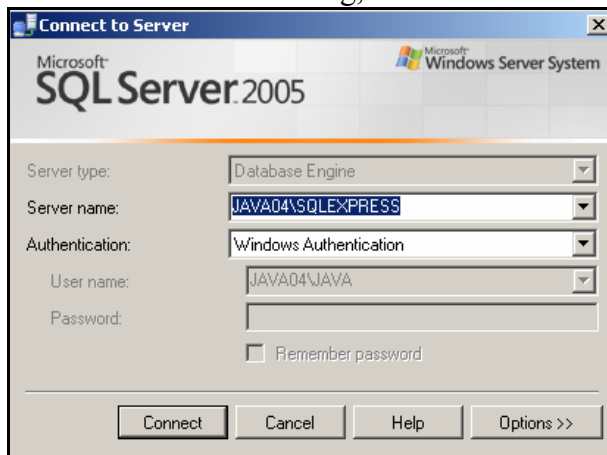
▪ A Main window will pop up.

▪ Notice that in the Main Window, the Object Explorer tab on the left lists all available
  elements of an SQL Server, such as available databases, security (logins and access
 controls, etc).
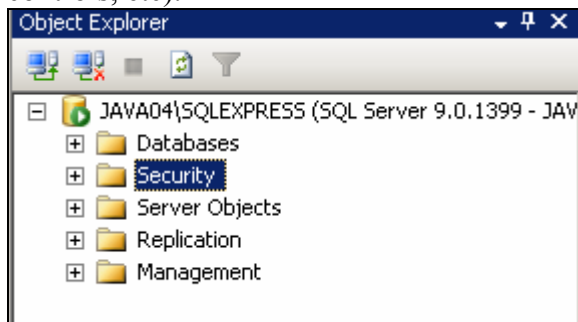
SQL Server Database Creation and Management

▪ Under Object Explorer, right click Databases and choose "New Database".
▪ Put in Database Name the value "CustomerOrder" and then press OK button.

Settings to allow User Connection to SQL Database

▪ Under the Object Explorer, "Right click" the Server SQLEXPRESS (named JAVAXX\SQLEXPRESS), and choose properties. [Note:XX being your computer name.]

▪ Choose the Security Tab. Make sure under the Server Authentication, "SQL Server and Windows Authentication" option is selected (to allow connection from JAVA program).
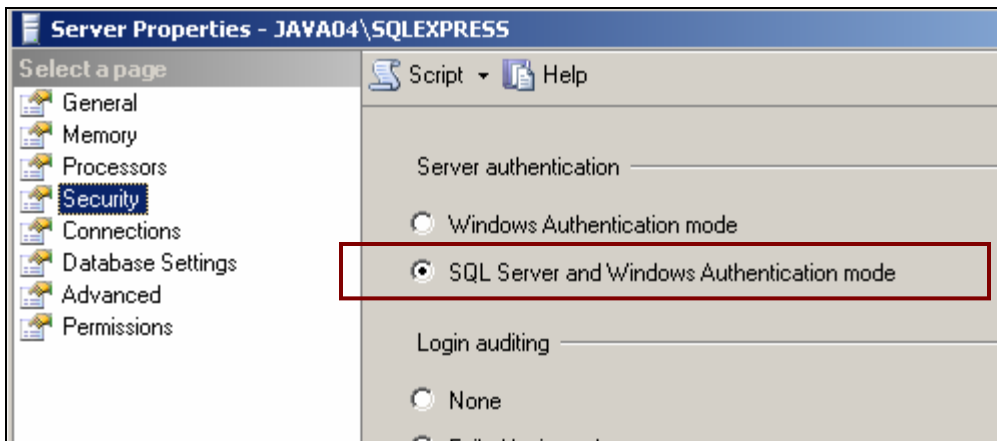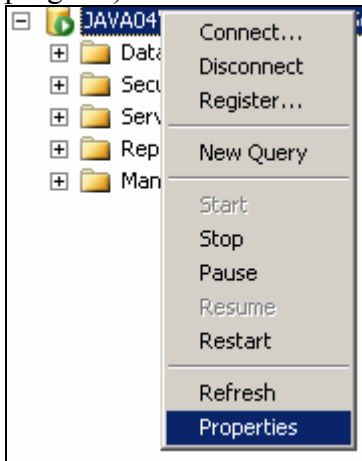
SQL Server User Account Creation and Management

A user account can be created on command line (sqlcmd) or by using the SQL Server Management software. To connect to SQL Server remotely or from programs like JAVA, we need to have one user account with appropriate access to the server and its database.

To create a user account with following details:
Username = student
Password = mypassword

- In the SQL Server Management software, get to the Logins main page.
- Right click and choose "New Login".
- Enter the Login name as "student".
- Choose "SQL Server Authentication" and enter the password.
- Make sure to check the "Enforce password policy" option.
- Make sure to uncheck the "Enforce password Expiration" option.
- Under User Mapping tab, associate the user student with Database "CustomerOrder" by checking the Map option.
- Under the Database Role list below the User Map list, the student should have the followings selected / checked:
 db_owner
  public

Alternatively, we can also use sqlcmd to create a user account.

- Connect to sqlcmd with windows authentication:
  Sqlcmd –S .\SQLEXPRESS
- Type the following below:
  > use CustomerOrder
  > go
  > create login student with password = 'mypassword'
  > go
  > create user student from login student
  > go
  > grant alter to student
  > grant control to student
  > go

- Now, we need to test if the user account is working:
- Under the SQL Server Management software, click File > Connect Object Explorer.
- The login window will pop out. Choose "SQL Server Authentication" as the authentication method and enter the username and password information for student.
- If nothing is wrong, then the account student will be able to log on.

- Alternatively, in command line (Start > run), you can try to type
  Sqlcmd –S .\SQLEXPRESS –U student –P mypassword
  You should be able to log in as student if all the steps above are followed correctly.

**Modifying User preferences and access to databases and tables**

Under the Object Explorer, expand the "Security" option. Left click "Logins" to list user accounts available (depending on your account type, server administrator sees all user accounts registered) in the SQL Server.

- To modify properties of a user, double click the name of that user.
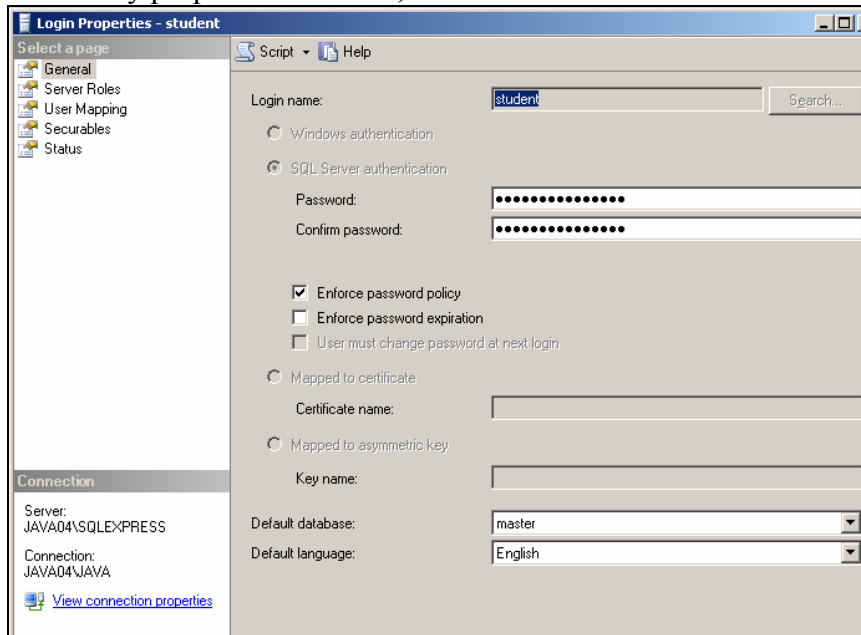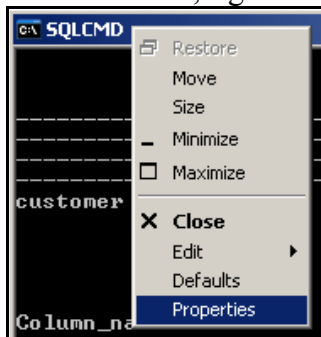

Tip: To improve display of databases and tables in sqlcmd

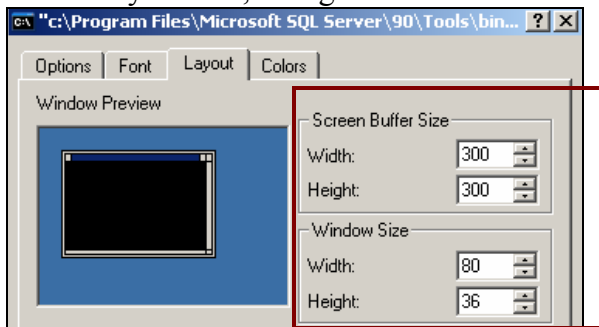Notice that it is very hard to read the table columns listing out properties as shown below:
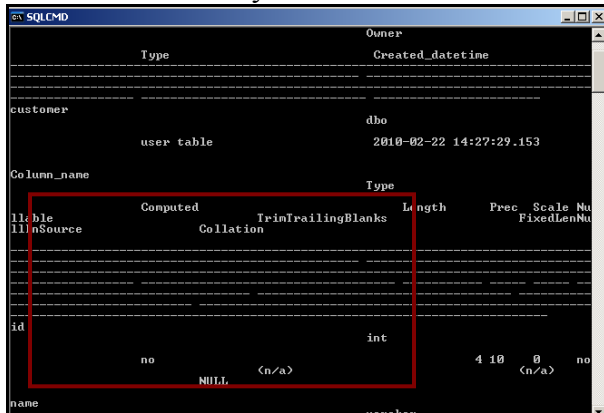
To correct this, right click the top bar of the command line window and select properties:
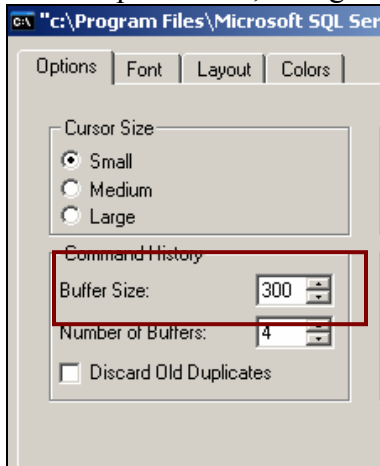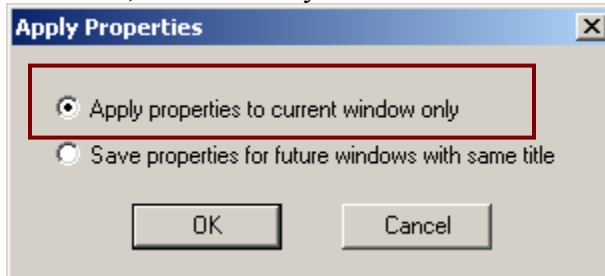
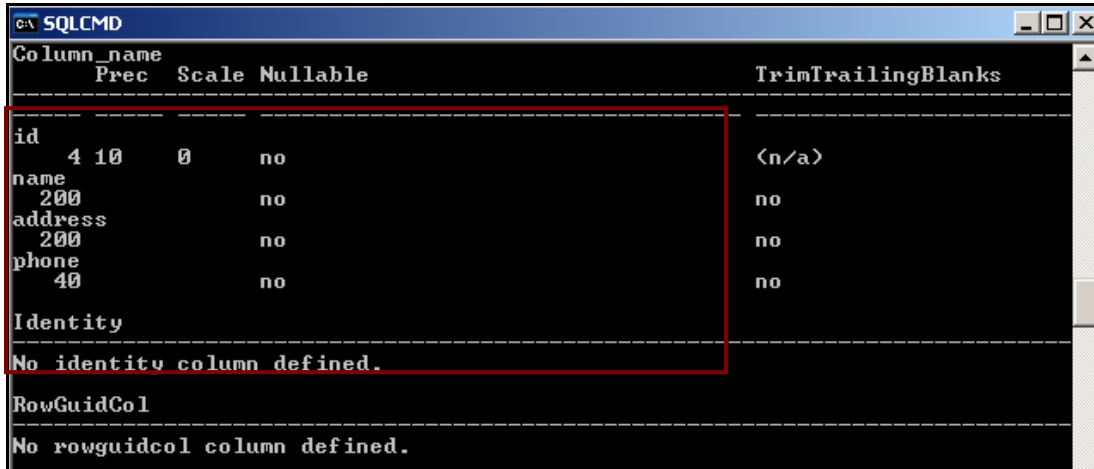Under Layout Tab, change the "Width" screen buffer size to about 5000.

Under Options Tab, change the "Buffer Size" to about 500.

Remember! Select "Apply properties to current window only". Otherwise, if you make a mistake, then it is very hard to reset the command window properties back to its default.

You will get a much sorted table listing out properties and such (as shown below):

**Integrating SQL Server 2005 Database with Java (using JDBC)**

Assuming that we have generated the java codes from VP for UML, we will now proceed with connecting the SQL database (SQL Server 2005) in our java programming.

- Start Eclipse IDE and set your preferred workspace. For example:
  - C:\JAVAProjects

- Unzip the zip file "DBJtds.zip" and put it in your preferred Java Eclipse working directory as stated above.

- The files after unzipping will contain 2 elements:
  - A folder "DBJtds" (This is the Java Project containing 6 java files)
  - A JAR file jtds-1.2.5.jar (which is the JDBC driver to SQL Server)

- To load the java project DBJtds, carefully follow the steps below:
  - Click File > New > Java Project.

  - In the New Java Project dialog box, choose "Create project from existing source" and specify the DBJtds working directory just now.

  - If required to put in a project name, just name it as DBJtds.
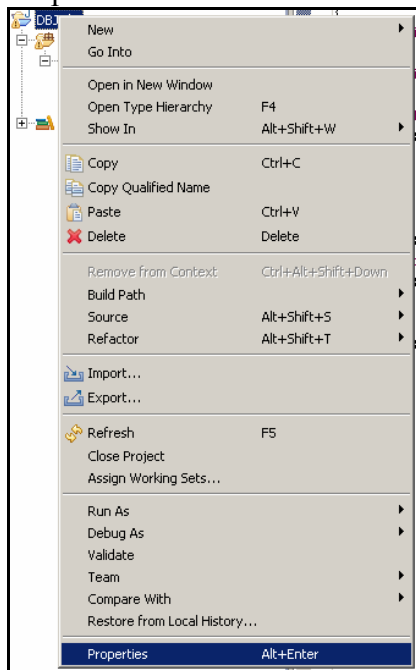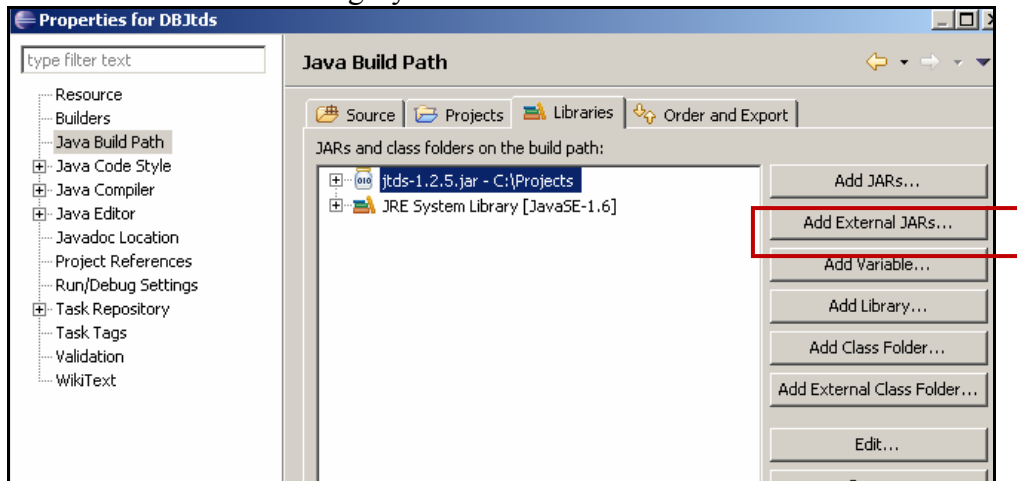
  - Then press "Finish".
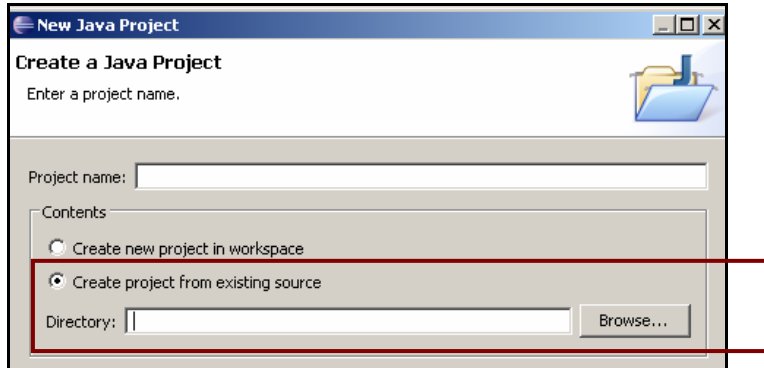
- When done correctly, you will get the following display in the Package Explorer.
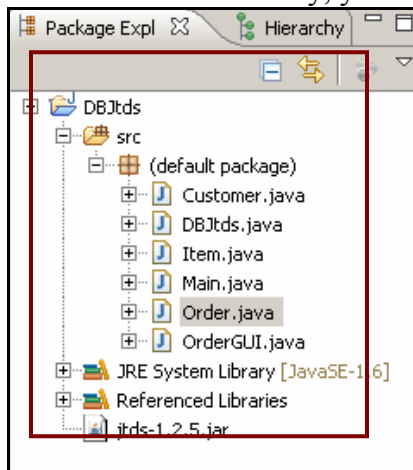
- Now, right click the DBJtds package name under the Package Explorer, and select Properties.

▪ Add the external JAR "jtds-1.2.5.jar" by pressing the "Add External JARs" button, under Java Build Path category.

▪ Now, we need to edit connection settings to SQL Server in the Java program. Open up Main.java in the Eclipse IDE (shown below).

```
1 public class Main
2 {
3        public Main() {}
4
5        public static void main (String args[])
6        throws Exception
7        {
8                // Change the values below according to your specs.
9                String database = "jdbc:jtds:sqlserver://localhost:1087/CustomerOrder";
10               String username = "kamal";
11               String password = "samsiah";
12
13               OrderGUI demo = new OrderGUI(database,username,password);
14               demo.initGUI();
15       }
16 }
```
Code snippet of Main.java

▪ Change the port number of the database to the one setup earlier in line 9:
"jdbc:jtds:sqlserver://localhost:**1087**/CustomerOrder"

▪ Also change the username and password if you want to use a different account (line 10 & 11).

▪ Save and run the JAVA program DBJtds -> Main.java by clicking Run Main in the toolbar, or pressing shortcut key "ctrl + F11".

You can add new values into the SQL table Customer by filling in all required inputs from the program. Try checking in between the values listed in the program and from sqlcmd or SSMSEE at the same time.

APPENDIX A – SOFTWARE REQUIRED FOR JAVA AND SQL DATABASE

Below listed are software and tools required for Java Database connectivity using SQL Server 2005 and Eclipse IDE (covered in this lab).

Software: Eclipse IDE
Description: IDE to code, debug and run Java programs.
Download Link: *http://www.eclipse.org/*

Software: Microsoft SQL Server 2005 Express Edition
Description: A free SQL Server by Microsoft. The latest version is actually Microsoft SQL Server 2008. Feel free to use that instead (not tested with this lab guide).
Download Link: *http://www.microsoft.com/Sqlserver/2005/en/us/express.aspx*

Software: Microsoft SQL Server Management Studio Express Edition (SSMSEE)
Description: A GUI front-end alternative to sqlcmd in managing SQL activities.
Download Link:
http://www.microsoft.com/downloads/details.aspx?FamilyID=c243a5ae-4bd1-4e3d-94b8-5a0f62bf7796&DisplayLang=en

Software: Jtds JDBC Driver
Description: An open source JDBC Driver to allow communication between Java application and SQL Server 2005. Actually, you only need the file jtds-1.2.5.jar from the zip file downloaded below.
Download Link: *http://jtds.sourceforge.net*