Lab Selection

Star-TED – V 2016.2.12.**143**

# Choose Your Lab Scenario!

This document contains over 30 distinct scenarios for simple CRUD operations on a modified version of the Adventure Works database (2005 edition).

You must select a single scenario for your lab project. Each student in the class must have a separate scenario (no two students are allowed to work on the same scenario). Before beginning your lab, you must sign up for your scenario and your selection must be approved by your instructor; contact your instructor for details. **If you are repeating the course, you must select a different scenario than the one you attempted in the previous term.**

For details on the lab specifications and the marking guide, please see the separate document titled “CPSC1517 Course Project”.

## Database Tables and Foreign Keys

Each scenario has an ERD (Entity Relationship Diagram) that shows the database tables for that scenario. Students are responsible to create the lab code for those tables in their scenario’s ERD; you only need to code for the tables that you actually use in your scenario.

For some scenarios, the tables will have Foreign Keys to other tables that are not in their ERD. For those situations, use the Free Code provided by your instructor when using DropDowns on your forms. The free code is available as a NuGet package at <https://www.nuget.org/packages/AWSystem.FreeCode>.

The namespaces for the classes are **FreeCode.AWSystem.BLL** and **FreeCode.AWSystem.Entities.POCOs**.

|  |  |  |
| --- | --- | --- |
|  |  |  |

# Heads-Up! – Stored Procedures

What’s in a name? A whole lot, actually. All of the tables and stored procedures in a database exist within something called a “database schema”. Most databases have just one schema, typically called “dbo”. The Adventure Works database, however, has five additional schemas.

Because of this, it is necessary to specify the schema name along with the table name as part of your Table() attribute on your entity classes. For example, if you have an entity for the Employee class, you would have to prefix the class with the following attribute: [Table("Employee", Schema = "HumanResources")]

Schema names also have to be included as part of any particular stored procedure that you wish to use. For example, if you wish to call the ***Employee\_ListAllManagers*** stored procedure, you will actually have to refer to it as “**HumanResources.*Employee\_ListAllManagers***” because it belongs to the **HumanResources** schema.

Here is a breakdown of the different schemas and their overall purpose in the Adventure Works database.

* **dbo** – The default schema name of “dbo” contains a small set of tables that have to do primarily with database auditing, such as dbo.AWBuildVersion, dbo.DatabaseLog and dbo.ErrorLog.
* **HumanResources** – This schema contains all of the employee-related elements of the Adventure Works business, such as HumanResources.Employee and HumanResources.Department.
* **Person** – This schema contains tables that store data that is typically people-specific, such as Person.Address, Person.Contact, etc. These tables relate to many different tables in the other schemas, often in a one-to-one relationship. For example, the HumanResources.Employee table has a foreign key called ContactID that relates to the Person.Contact table. Additionally, Sales.Individual and Sales.StoreContact also have foreign keys related to Person.Contact.
* **Production** – In this schema, all of the tables relate to inventory, such as Production.Product, Production.BillOfMaterials, Production.WorkOrder, etc.
* **Purchasing** – All of the tables in this schema relate to information about external agencies through whom inventory is purchased, such as Purchasing.Vendor and Purchasing.ShipMethod
* **Sales** – Since Adventure Works is a company that deals primarily with selling products, this schema is dedicated to tracking and maintaining all information related to Sales, such as Sales.Customer, Sales.CreditCard, Sales.SpecialOffer, etc.

# Use this web address to review the contents of AdventureWorks tables.

http://elsasoft.com/samples/sqlserver\_adventureworks/SqlServer.SPRING.KATMAI.AdventureWorks/allTables.htm

|  |  |
| --- | --- |
|  | A1: ScenarioYou must do three forms (A, B and C).  * **Form A** - Single Item Create/Read/Update/Delete   + Table * **Form B** - Gridview Lookup with Code-Behind   + Table by Table * **Form C** - Gridview Lookup with ObjectDataSource controls   + Table by Table  Recommended Stored Procedures: The following specialty stored procedures are available:   * **Sproc**  Important Notes:  * **Employee\_Delete** is not available, because employees can only be marked as “current” or “not current” (**CurrentFlag**). |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |



Course Project Selection

Signup Sheet

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Human Resources** | |  | **Production** | |
| **H1** |  | **P1** |  |
| **H2** |  | **P2** |  |
| **H3** | **Not Available** | **P3** |  |
| **H4** |  | **P4** |  |
| **H5** |  | **P5** |  |
| **H6** |  | **P6** |  |
| **Sales** | | **P7 \*** |  |
| **S1** |  | **P8 \*** |  |
| **S2** |  | **P9** |  |
| **S3** |  | **P10** |  |
| **S4** |  | **Purchasing** | |
| **S5** |  | **R1** |  |
| **S6** |  | **R2** |  |
| **S7** |  | **R3** |  |
| **S8 \*** |  | **R4** |  |
| **S9** |  | **R5** |  |
| **S10** |  | **R6** |  |
| **S11** |  |  |  |

\* This selection is rated Difficult and should only be attempted if approved by your instructor