­

|  |
| --- |
| ***Computer Engineering Department*** |
| ***CE302L: Database Management Systems*** |

|  |  |
| --- | --- |
| ***Course Instructor: Umer Farooq*** | ***Dated: January 24, 2022*** |
| ***Lab Engineer: Muhammad Usama Riaz*** | ***Semester: 5th*** |
| ***Session: 2019-2023*** | ***Batch: BSCE2019*** |

# Lab. 12 User Defined Functions

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Roll No** | **Lab Report Marks/100** | **Total Marks**  **(Scaled out of 10)** |
|  |  |  |  |

Checked on:

Signature:

## 12.1 Objective

## We will also study User defined functions, it’s comparison with stored procedures and views.

## 12.2 Equipment and Component

|  |  |  |
| --- | --- | --- |
| **Component Description/Software** | **Value** | **Quantity** |
| Computer, MS SQL Server | On Campus | 1 |

## 12.3 Conduct of Lab

* + 1. Students are required to perform this experiment individually.
    2. In case the lab experiment is not understood, the students are advised to seek help from the course instructor, lab engineers, assigned teaching assistants (TA) and lab attendants.

## 12.4 Theory and Background

**User defined functions**

User defined functions are routines that encapsulates SQL logic inside it. Like stored procedures User defined functions can also be passed input parameters but user defined functions are compiled and executed at runtime. Therefore, they are slower than stored procedures.

|  |
| --- |
| Syntax:  CREATE FUNCTION dbo.Function      (      /\*      @parameter1 datatype = default value,      @parameter2 datatype      \*/      )   RETURNS /\* datatype \*/   AS      BEGIN       /\* sql statement ... \*/      RETURN/\* value \*/      END  **Certain limitations for User defined functions:**   1. UDF can’t perform DML (data manipulation language) operations like Insertion, Update and Deletion on the base table. 2. UDF can’t return non deterministic values like GETDATE () etc. 3. Stored procedure can’t be called from inside a UDF whereas a stored procedure can call a user defined function or another stored procedure inside it.   There are two types of user defined functions:   * Scalar * Inline  Scalar Functions (returns a single value) Example:  CREATE FUNCTION EmployeeContactID(@Empid int)  RETURNS int  AS  BEGIN  Declare @returnvalue int  Select @returnvalue=Employee.ContactID from Employee where Employee.EmployeeID=@Empid  RETURN @returnvalue  END  **Execution:**  select dbo.EmployeeContactID(1) Inline Functions (returns a table) Example:  CREATE FUNCTION dbo.GetEmployeeFunction(@empid int)  RETURNS TABLE  AS  RETURN SELECT \*  FROM employee where employee.EmployeeID=@empid  **Execution:**  select\*from dbo.GetEmployeeFunction(1) |

## Difference between procedures and user defined functions

1. A stored procedure is precompiled while a User defined function is compiled and executed at runtime.
2. A Stored procedure is more flexible than user defined function like you can write complex logic (for example exceptional handling using try catch block is possible in stored procedures which is not possible in user defined functions)
3. A stored procedure can call another stored procedure or user defined function inside it but a user defined function can’t call stored procedure inside it.
4. A stored procedure can return nondeterministic values but a user defined function can’t return a non-deterministic value like Get Date () function.
5. A user defined functions does not support DML operations like insertion, deletion and update on the base table but it is possible via stored procedure.
6. A user defined function is easier to execute and can be used inside selection and even for joins but stored procedure can’t be used inside selection queries and it can’t be used to join with other tables.

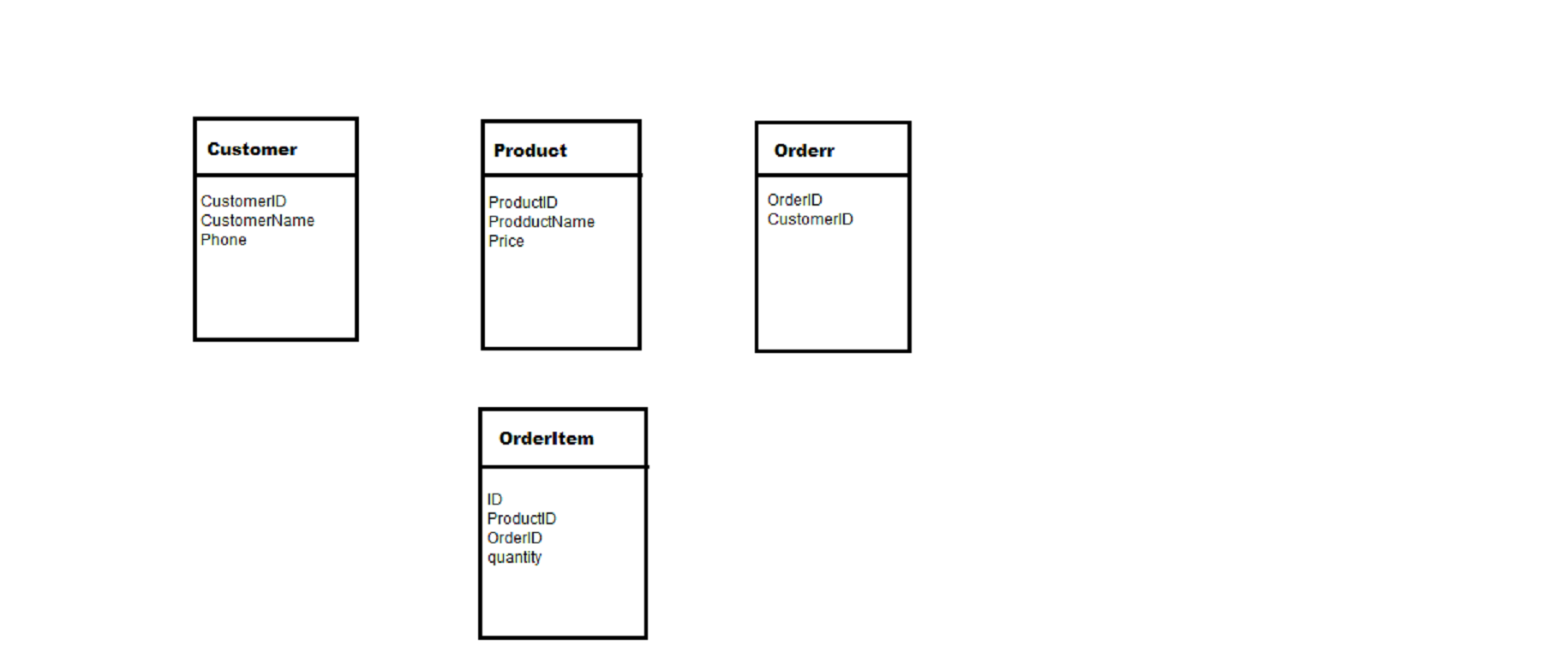
## 

## Some comparison with views

If you think of view than a question might arise in your mind, why we don’t use views instead of stored procedures or user defined functions for basic SQL selection queries. Answer is flexibility. You can’t pass parameters to views for selection of filtered queries, but stored procedures and user defined functions provide you these features. Similarly, Multiple DML operations are restricted in views which are possible through stored procedures and user defined functions.

## 12.5 Lab Task:

1). Use the Customer Product relational database in file ‘InLab\_Script’ to attempt the questions below.



* 1. Write a scalar UDF that calculates the total sale(money) of a certain product.
  2. Write an inline UDF that lists the information of all the ordered products by a specific user, the user id is passed as input parameter.
  3. Write a procedure which displays productID, productName of all products along with total sale of that product.(Hint: use function created in 1.1).

#### Assessment Rubrics for Lab

Method:Lab reports and instructor observation during lab sessions.

**Outcome assessed:**

a. Ability to conduct experiments, as well as to analyze and interpret data (P)

b. Ability to function on multi-disciplinary teams (A)

c. Ability to use the techniques, skills, and modern engineering tools necessary for engineering practice (P)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Performance metric | Mapping (task no. and description) | | Max marks | Exceeds expectation | Meets expectation | Does not meet expectation | Obtained marks |
| 1. Realization of experiment (a) | 1 | Functionality | 40 | Executes without errors excellent user prompts, good use of symbols, spacing in output. Through testing has been completed (45-41) | Executes without errors, user prompts are understandable, minimum use of symbols or spacing in output. Some testing has been completed (40-21) | Does not execute due to syntax errors, runtime errors, user prompts are misleading or non-existent. No testing has been completed (20-0) |  |
| 2. Teamwork (b) | 1 | Group Performance | 5 | Actively engages and cooperates with other group member(s) in effective manner (5-4) | Cooperates with other group member(s) in a reasonable manner but conduct can be improved (3-2) | Distracts or discourages other group members from conducting the experiment (1-0) |  |
| 3. Conducting experiment (a, c) | 1 | On Spot Changes | 10 | Able to make changes (5-4) | Partially able to make changes (3-2) | Unable to make changes (1-0) |  |
| 2 | Viva | 10 | Answered all questions (5-4) | Few incorrect answers (3-2) | Unable to answer all questions (1-0) |  |
| 4. Laboratory safety and disciplinary rules (a) | 1 | Code commenting | 5 | Observes lab safety rules; adheres to the lab disciplinary guidelines aptly (5-4) | Generally, observes safety rules and disciplinary guidelines with minor lapses (3-2) | Disregards lab safety and disciplinary rules (1-0) |  |
| 5. Data collection (c) | 1 | Code Structure | 5 | Excellent use of white space, creatively organized work, excellent use of variables and constants, correct identifiers for constants, No line-wrap (5-4) | Includes name, and assignment, white space makes the program fairly easy to read. Title, organized work, good use of variables (3-2) | Poor use of white space (indentation, blank lines) making code hard to read, disorganized and messy (1-0) |  |
| 6. Data analysis (a, c) | 1 | Algorithm | 20 | Solution is efficient, easy to understand, and maintain (5-4) | A logical solution that is easy to follow but it is not the most efficient (3-2) | A difficult and inefficient solution (1-0) |  |
| 7. Computer use (c) | 1 | Documentation | 5 | Timely documented (5-4) | Late documented (3-2) | Not documented (1-0) |  |
|  | Max Marks (total): | | 100 | Obtained Marks (Total): | | |  |

Lab Engineer Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_