

# Database Basics MS SQL Exam – 20 Oct 2019

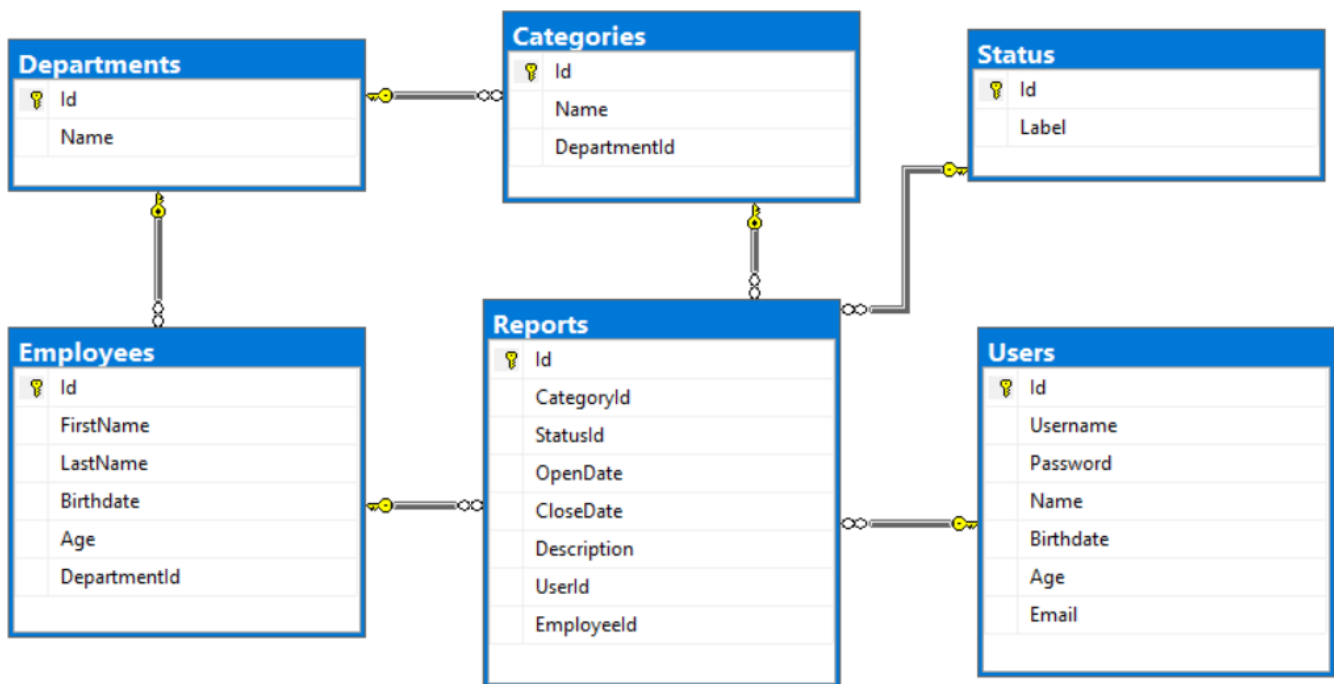
Exam problems for the [“Database Basics MSSQL Server” course @ SoftUni](#) Submit your solutions in the SoftUni judge system at Software University.

## Service

The city mayor, came up with the idea to create an online platform where all the citizens can **report about different problems** and a special organization will work to resolve all the incoming reports. This organization has a few **departments each of which is responsible for a set of problem's categories** in which **users can submit a report**. In each department there are employees who get assigned to a report. Of course, this huge platform needs a reliable database to store and process the information and the mayor has asked for the best specialist in this area. That's why you got chosen! Congratulations and good luck!

## Section 1. DDL (30 pts)

You have been given the E/R Diagram of the Report Service:



Create a database called **Service**. You need to create **6 tables**:

- **Users** - contains information about the people who submit reports
- **Reports** - contains information about the problems
- **Employees** - contains information about the employees
- **Departments** - contains information about the departments
- **Categories** - contains information about categories in reports.

- **Status** - contains information about the possible

## Users

Column Name	Data Type	Constraints
Id	Integer from <b>0 to 2,147,483,647</b>	<b>Unique</b> table identifier, <b>Identity</b> .
Username	String up to <b>30 symbols</b>	<b>Unique</b> for each user, <b>NULL</b> is <b>NOT</b> permitted.
Password	String up to <b>50 symbols</b>	<b>NULL</b> is <b>NOT</b> permitted.
Name	String up to <b>50 symbols</b>	<b>NULL</b> is permitted.
Birthdate	Date <b>with time</b>	<b>NULL</b> is permitted.
Age	Integer from <b>0 to 2,147,483,647</b>	In range between <b>14</b> and <b>110 (inclusive)</b> .
Email	String up to <b>50 symbols</b>	<b>NULL</b> is <b>NOT</b> permitted.

## Departments

Column Name	Data Type	Constraints
Id	Integer from <b>0 to 2,147,483,647</b>	<b>Unique</b> table identifier, <b>Identity</b> .
Name	String up to 50 symbols	<b>NULL</b> is <b>NOT</b> permitted.

## Employees

Column Name	Data Type	Constraints
Id	Integer from <b>0 to 2,147,483,647</b>	<b>Unique</b> table identifier, <b>Identity</b> .
FirstName	String up to <b>25 symbols</b>	<b>NULL</b> is permitted.
LastName	String up to <b>25 symbols</b>	<b>NULL</b> is permitted.
Birthdate	Date <b>with time</b>	<b>NULL</b> is permitted.
Age	Integer from <b>0 to 2,147,483,647</b>	In range between <b>18</b> and <b>110 (inclusive)</b> .
DepartmentId	Integer from <b>0 to 2,147,483,647</b>	Relationship with table <b>departments</b> .

## Categories

Column Name	Data Type	Constraints
Id	Integer from <b>0 to 2,147,483,647</b>	<b>Unique</b> table identifier, <b>Identity</b> .
Name	String up to <b>50 symbols</b>	<b>NULL</b> is <b>NOT</b> permitted.
DepartmentId	Integer from <b>0 to 2,147,483,647</b>	Relationship with table <b>departments</b> . <b>NULL</b> is <b>NOT</b> permitted.

## Status

Column Name	Data Type	Constraints
Id	Integer from <b>0 to 2,147,483,647</b>	<b>Unique</b> table identifier, <b>Identity</b> .
Label	String up to <b>30 symbols</b>	<b>NULL</b> is <b>NOT</b> permitted.

## Reports

Column Name	Data Type	Constraints
Id	Integer from 0 to 2,147,483,647	Unique table identifier, Identity.
CategoryId	Integer from 0 to 2,147,483,647	Relationship with table <b>categories</b> . NULL is NOT permitted.
StatusId	Integer from 0 to 2,147,483,647	Relationship with table <b>status</b> . NULL is NOT permitted.
OpenDate	Date with time	NULL is NOT permitted.
CloseDate	Date with time	NULL is permitted.
Description	String up to 200 symbols	NULL is NOT permitted.
UserId	Integer from 0 to 2,147,483,647	Relationship with table <b>users</b> . NULL is NOT permitted.
EmployeeId	Integer from 0 to 2,147,483,647	Relationship with table <b>employees</b> .

## 1. Table design

Submit all of your **create statements** to Judge.

## Section 2. DML (10 pts)

**Before you start you have to import "DataSet-Service.sql". If you have created the structure correctly the data should be successfully inserted.**

In this section, you have to do some data manipulations:

## 2. Insert

Let's **insert** some sample data into the database. Write a query to add the following records into the corresponding tables. All Id's should be auto-generated.

### Employees

FirstName	LastName	Birthdate	DepartmentId
Marlo	O'Malley	1958-9-21	1
Niki	Stanaghan	1969-11-26	4
Ayrton	Senna	1960-03-21	9
Ronnie	Peterson	1944-02-14	9
Giovanna	Amati	1959-07-20	5

## Reports

CategoryId	StatusId	OpenDate	CloseDate	Description	UserId	EmployeeId
1	1	2017-04-13		Stuck Road on Str.133	6	2
6	3	2015-09-05	2015-12-06	Charity trail running	3	5
14	2	2015-09-07		Falling bricks on Str.58	5	2
4	3	2017-07-03	2017-07-06	Cut off streetlight on Str.11	1	1

### 3. Update

Update the CloseDate with the current date of all reports, which don't have CloseDate.

### 4. Delete

Delete all reports who have a Status 4.

## Section 3. Querying (40 pts)

You need to start with a fresh dataset, so recreate your DB and import the sample data again (DataSet-Service.sql).

### 5. Unassigned Reports

Find all reports that don't have an assigned employee. Order the results by OpenDate in ascending order, then by description ascending. OpenDate must be in format - 'dd-MM-yyyy'

**Example:**

Description	OpenDate
Art exhibition on July 24	17-12-2014
Stuck Road on Str.133	20-06-2015
Burned facade on Str.560	26-08-2015

### 6. Reports & Categories

Select all descriptions from reports, which have category. Order them by description (ascending) then by category name (ascending).

**Example:**

Description	CategoryName
162 kg plastic for recycling.	Green Areas
246 kg plastic for recycling.	Snow Removal
366 kg plastic for recycling.	Recycling

### 7. Most Reported Category

Select the top 5 most reported categories and order them by the number of reports per category in descending order and then alphabetically by name.

**Example:**

CategoryName	ReportsNumber
Recycling	8
Snow Removal	5
Streetlight	4

## 8. Birthday Report

Select the user's **username** and **category name** in all **reports** in which **users** have submitted a report **on their birthday**. Order them by **username (ascending)** and then by **category name (ascending)**.

### Example:

Username	CategoryName
5omarkwelleyc	Snow Removal
dpennid	Dangerous Trees
llankham6	Homeless Elders

## 9. Users per Employee

Select **all employees** and show how many **unique** users each of them has served to.

Order by **users count (descending)** and then by **full name (ascending)**.

### Example:

FullName	UsersCount
Bron Ledur	3
Adelind Benns	2
Dick Wentworth	2
...	...

## 10. Full Info

Select **all info** for reports along with employee **first name** and **last name (concatanated with space)**, **department name**, **category name**, **report description**, **open date**, **status label** and **name of the user**. Order them by first name (**descending**), last name (**descending**), department (**ascending**), category (**ascending**), description (**ascending**), open date (**ascending**), status (**ascending**) and user (**ascending**).

Date should be in format - **dd.MM.yyyy**

If there are empty records, replace them with '**None**'.

### Example:

Employee	Department	Category	Description	OpenDate	Status	User
Niki Stranaghan	Event Management	Sports Events	Sky Run competition on September 8	08.06.2015	Completed	Emlynn Alliberton
Marlo O'Malley	Infrastructure	Streetlight	Fallen streetlight columns on Str.14	12.09.2017	Blocked	Erhart Alpine
Leonardo Shopcott	Animals Care	Animal in Danger	Parked car on green area on the sidewalk of Str.74	10.11.2016	In Progress	Jocko Greggor
...	...	...	...	...	...	...

## Section 4. Programmability (20 pts)

### 11. Hours to Complete

Create a **user defined function** with the name `udf_HoursToComplete(@StartDate DATETIME, @EndDate DATETIME)` that receives a start date and end date and must returns the total hours which has been taken for this task. If start date is null or end is null return 0.

#### Example usage:

Query
<pre>SELECT dbo.udf_HoursToComplete(OpenDate, CloseDate) AS TotalHours FROM Reports</pre>
TotalHours
0
120
0

### 12. Assign Employee

Create a **stored procedure** with the name `usp_AssignEmployeeToReport(@EmployeeId INT, @ReportId INT)` that receives an **employee's Id** and a **report's Id** and assigns the employee to the report **only if** the department of the employee and the department of the report's category are the same. Otherwise throw an **exception** with message: "**Employee doesn't belong to the appropriate department!**".

#### Example usage:

Query
<pre>EXEC usp_AssignEmployeeToReport 30, 1</pre>
Response
Employee doesn't belong to the appropriate department!
Query
<pre>EXEC usp_AssignEmployeeToReport 17, 2</pre>
Response
(1 row affected)