# Exercises: Data Definition and Data Types

This document defines the **exercise assignments** for the ["PHP Web Development Basics" course @ Software University.](https://softuni.bg/trainings/2163/php-web-development-basics-september2018#lesson-9664)

You can check your solutions here: <https://judge.softuni.bg/Contests/1274/Introduction-to-MySQL-Exercise>

## Create Database

**Create new table** named **minions(id, name, age, town\_id)**. Then add new table **towns (id, name).** Set **id** columns of both tables to be **primary key** as **constraint**. Submit your create table queries in Judge together for both tables (one after another separated by “;”) as **Run queries & check DB.**

## Insert Records in Both Tables

**Populate both tables** with sample records given in the table below.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **minions** | | | |  | **towns** | |
| **id** | **name** | **age** | **town\_id** |  | **id** | **name** |
| 1 | Kevin | 22 | 1 |  | 1 | Sofia |
| 2 | Bob | 15 | 3 |  | 2 | Plovdiv |
| 3 | Steward | NULL | 2 |  | 3 | Varna |

Use only insert SQL queries. Submit your **INSERT statements** in Judge as **Run skeleton, run queries & check DB**.

## Truncate Table Minions

**Delete all the data** from the **minions** table using **SQL query.** Submit your query in Judge as **Run skeleton, run queries & check DB**.

## Drop All Tables

**Delete all tables** from the **minions** database using **SQL query**.Submit your query in Judge as **Run skeleton, run queries & check DB**.

## \*Create Table People

Using **SQL query** create table “**people**” with columns:

* **id** – unique number for every person there will be **no more than 231**-1**people.** (Auto incremented)
* **name** – full name of the person will be **no more than 200 Unicode characters**. (Not null)
* **picture** – image with **size up to** **2 MB.** (Allow nulls)
* **height** – In meters. Real number precise up to **2 digits** after floating point. (Allow nulls)
* **weight** – In kilograms. Real number precise up to **2 digits** after floating point. (Allow nulls)
* **gender** – Possible states are **m** or **f.** (Not null)
* **birthdate –** (Not null)
* **biography** – detailed biography of the person it can contain **max allowed Unicode characters.** (Allow nulls)

Make **id** primary key. Populate the table with **5 records**. Submit your **CREATE** and **INSERT statements** in Judge as **Run queries & check DB**.

## \*Create Table Users

Using **SQL query** create table **users** with columns:

* **id** – unique number for every user. There will be **no more than 263-1 users.** (Auto incremented)
* **username** – unique identifier of the user will be **no more than 30 characters (non Unicode).** (Required)
* **password** – password will be **no longer than 26 characters (non Unicode).** (Required)
* **profile\_picture** – image with **size up to 900 KB.**
* **last\_login\_time**
* **is\_deleted** – shows if the user deleted his/her profile. Possible states are **true** or **false**.

Make **id** primary key. Populate the table with **5 records**. Submit your **CREATE** and **INSERT statements**. Submit your **CREATE** and **INSERT statements** as **Run queries & check DB.**

## \*Change Primary Key

Using **SQL queries** modify table **users** from the previous task. First **remove current primary key** then create **new primary key** that would be **combination** of fields **id** and **username**. The initial primary key name on **id** is **pk\_users**. Submit your query in Judge as **Run skeleton, run queries & check DB**.

# Part I – Queries for SoftUni Database

Download and get familiar with the **soft\_uni**, **diablo** and **geography** database schemas and tables. You will use them in this and the following exercises to write queries.

## 8.Find All Information About Departments

Write a SQL query to find **all available information about the departments. Sort the information by id.** Submit your query statements as **Prepare DB & run queries.**

### Example

|  |  |  |
| --- | --- | --- |
| **department\_id** | **name** | **manager\_id** |
| 1 | Engineering | 12 |
| 2 | Tool Design | 4 |
| 3 | Sales | 273 |
| … | … | … |

## 9.Find all Department Names

Write SQL query to find **all department names**. **Sort the information by id.** Submit your query statements as **Prepare DB & run queries.**

### Example

|  |
| --- |
| **name** |
| Engineering |
| Tool Design |
| Sales |
| … |

## 10.Find salary of Each Employee

Write SQL query to find the **first name**, **last name** and **salary** of each employee. **Sort the information by id.** Submit your query statements as **Prepare DB & run queries.**

### Example

|  |  |  |
| --- | --- | --- |
| **first\_name** | **last\_name** | **salary** |
| Guy | Gilbert | 12500.00 |
| Kevin | Brown | 13500.00 |
| Roberto | Tamburello | 43300.00 |
| … | … | … |

## 11.Find Full Name of Each Employee

Write SQL query to find the **first**, **middle** and **last name** of each employee. **Sort the information by id.** Submit your query statements as **Prepare DB & run queries.**

### Example

|  |  |  |
| --- | --- | --- |
| **first\_name** | **middle\_name** | **last\_name** |
| Guy | R | Gilbert |
| Kevin | F | Brown |
| Roberto | NULL | Tamburello |
| … | … | … |

## 12.Find Email Address of Each Employee

Write a SQL query to find the **email address** of each employee. (by his **first and last name**). Consider that the email domain is **softuni.bg**. Emails should look like “John.Doe@softuni.bg". The **produced column** should be named   
**"full\_ email\_address"**. Submit your query statements as **Prepare DB & run queries**.

### Example

|  |
| --- |
| **full\_email\_address** |
| Guy.Gilbert@softuni.bg |
| Kevin.Brown@softuni.bg |
| Roberto.Tamburello@softuni.bg |
| … |

## 13.Find All Different Employee’s Salaries

Write a SQL query to find **all different employee’s salaries**. Show only the salaries. **Sort the information by salary.**  Submit your query statements as **Prepare DB & run queries**.

### Example

|  |
| --- |
| **Salary** |
| 9000.00 |
| 9300.00 |
| 9500.00 |
| … |

## 14.Find all Information About Employees

Write a SQL query to find **all information** about the employees whose **job title** is **“Sales Representative”. Sort the information by id.** Submit your query statements as **Prepare DB & run queries**.

### Example

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **id** | **First**  **Name** | **Last**  **Name** | **Middle**  **Name** | **Job Title** | **DeptID** | **Mngr**  **ID** | **HireDate** | **salary** | **address\_id** |
| 275 | Michael | Blythe | G | Sales Representative | 3 | 268 | … | 23100.00 | 60 |
| 276 | Linda | Mitchell | C | Sales Representative | 3 | 268 | … | 23100.00 | 170 |
| 277 | Jillian | Carson | NULL | Sales Representative | 3 | 268 | … | 23100.00 | 61 |
| … | … | … | … | … | … | … | … | … | … |

## 15.Find Names of All Employees by salary in Range

Write a SQL query to find the **first name**, **last name** and **job title** of all employees whose **salary is in the** **range [20000, 30000]. Sort the information by id.** Submit your query statements as **Prepare DB & run queries**.

### Example

|  |  |  |
| --- | --- | --- |
| **first\_name** | **last\_name** | **JobTitle** |
| Rob | Walters | Senior Tool Designer |
| Thierry | D'Hers | Tool Designer |
| JoLynn | Dobney | Production Supervisor |
| … | … | … |

## 16. Find Names of All Employees

Write a SQL query to find the **full name** of all employees whose **salary** is **25000, 14000, 12500 or 23600**. Full Name is combination of **first**, **middle** and **last** name (separated with **single space**) and they should be **in one column** called **“Full Name”.** Submit your query statements as **Prepare DB & run queries**.

### Example

|  |
| --- |
| **full\_name** |
| Guy R Gilbert |
| Thierry B D'Hers |
| JoLynn M Dobney |

## 17.Find All Employees Without Manager

Write a SQL query to find **first and last names** about those employees that **does not have a manager**. Submit your query statements as **Prepare DB & run queries**.

### Example

|  |  |
| --- | --- |
| **first\_name** | **last\_name** |
| Ken | Sanchez |
| Svetlin | Nakov |
| … | … |

## 18.Find All Employees with salary More Than 50000

Write a SQL query to find **first name**, **last name** and **salary** of those employees who has salary more than 50000. Order them in decreasing order by salary. Submit your query statements as **Prepare DB & run queries**.

### Example

|  |  |  |
| --- | --- | --- |
| **first\_name** | **last\_name** | **salary** |
| Ken | Sanchez | 125500.00 |
| James | Hamilton | 84100.00 |
| … | … | … |

## 19.Find 5 Best Paid Employees

Write SQL query to find **first and last names** about **5 best paid Employees** ordered **descending by their salary.** Submit your query statements as **Prepare DB & run queries**.

### Example

|  |  |
| --- | --- |
| **first\_name** | **last\_name** |
| Ken | Sanchez |
| James | Hamilton |
| … | … |

## 20.Find All Employees Except Marketing

Write a SQL query to find the **first** and **last names** of all employees whose **department ID is different from 4.** Submit your query statements as **Prepare DB & run queries.**

### Example

|  |  |
| --- | --- |
| **first\_name** | **last\_name** |
| Guy | Gilbert |
| Roberto | Tamburello |
| Rob | Walters |

## 21.Sort Employees Table

Write a SQL query to sort all records in the **еmployees** table by the following criteria:

* First by **salary** in **decreasing** order
* Then by **first name** **alphabetically**
* Then by **last name descending**
* Then by **middle name alphabetically**

**Sort the information by id.** Submit your query statements as **Prepare DB & run queries.**

### Example

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **id** | **First**  **Name** | **Last**  **Name** | **Middle**  **Name** | **job\_title** | **DeptID** | **Mngr**  **ID** | **HireDate** | **salary** | **address\_id** |
| 109 | Ken | Sanchez | J | Chief Executive Officer | 16 | NULL | … | 125500.00 | 177 |
| 148 | James | Hamilton | R | Vice President of Production | 7 | 109 | … | 84100.00 | 158 |
| 273 | Brian | Welcker | S | Vice President of Sales | 3 | 109 | … | 72100.00 | 134 |
| … | … | … | … | … | … | … | … | … | … |

## 22.Distinct Job Titles

Write a SQL query to find **all distinct job titles**. Submit your query statements as **Prepare DB & run queries.** Ordered **ascending by their job\_title**

### Example

|  |
| --- |
| **Job\_title** |
| Accountant |
| Accounts Manager |
| Accounts Payable Specialist |
| … |

## 23.Find First 10 Started Projects

Write a SQL query to find **first 10 started projects**. Select **all information about them** and **sort** them **by start date**, **then by name**. **Sort the information by id.**  Submit your query statements as **Prepare DB & run queries.**

### Example

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **id** | **Name** | **Description** | **start\_date** | **end\_date** |
| 6 | HL Road Frame | Research, design and development of HL Road … | 1998-05-02 00:00:00 | 2003-06-01 00:00:00 |
| 2 | Cycling Cap | Research, design and development of C… | 2001-06-01 00:00:00 | 2003-06-01 00:00:00 |
| 5 | HL Mountain Frame | Research, design and development of HL M… | 2001-06-01 00:00:00 | 2003-06-01 00:00:00 |
| … | … | … | … | … |

## 24. Last 7 Hired Employees

Write a SQL query to find **last 7 hired employees**. Select **their first, last name and their hire date.** Submit your query statements as **Prepare DB & run queries.**

### Example

|  |  |  |
| --- | --- | --- |
| **first\_name** | **last\_name** | **hire\_date** |
| Rachel | Valdez | 2005-07-01 00:00:00 |
| Lynn | Tsoflias | 2005-07-01 00:00:00 |
| Syed | Abbas | 2005-04-15 00:00:00 |
| … | … | … |

## 25.\*Increase Salaries

Write a SQL query to increase salaries of all employees that are in the **Engineering**, **Tool Design**, **Marketing** or **Information Services** department by **12%**. Then **select Salaries column** from the **Employees** table. Submit your query statements as **Prepare DB & run queries.**

### Example

|  |
| --- |
| **Salary** |
| 12500.00 |
| 15120.00 |
| 48496.00 |
| 33376.00 |
| … |

# Part II – Queries for Geography Database

## 26. All Mountain Peaks

Display all **mountain peaks** in alphabetical order. Submit your query statements as **Prepare DB & run queries.**

### Example

|  |
| --- |
| **peak\_name** |
| Aconcagua |
| Banski Suhodol |
| Batashki Snezhnik |
| … |

## 27. Biggest Countries by Population

Find the **30 biggest countries** by **population** from **Europe**. Display the **country name** and **population**. Sort the results by **population** **(from biggest to smallest),** then by **country** **alphabetically**. Submit your query statements as **Prepare DB & run queries.**

### Example

|  |  |
| --- | --- |
| **country\_name** | **population** |
| Russia | 140702000 |
| Germany | 81802257 |
| France | 64768389 |
| … | … |

## 28. \*Countries and Currency (Euro / Not Euro)

Find all countries along with information about their *currency*. Display the **country** **name**, **country code** and information about its **currency**: either "**Euro**" or "**Not Euro**". Sort the results by **country name alphabetically**. Submit your query statements as **Prepare DB & run queries.**

### Example

|  |  |  |
| --- | --- | --- |
| **country\_name** | **country\_code** | **currency** |
| Afghanistan | AF | Not Euro |
| Åland | AX | Euro |
| Albania | AL | Not Euro |
| … | … | … |