**Exercises: Basic CRUD**

This document defines the **exercise assignments** for the [MySQL course @ Software University.](https://softuni.bg/opencourses/databases-basics-mysql)

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.

# Part I – Queries for SoftUni Database

## 1. 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.**

**select \* from `departments`**

**order by `department\_id`;**

### Example

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

## 2. 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.**

**select `name` from `departments`**

**order by `department\_id`;**

### Example

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

## 3. 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.**

**select `first\_name`,`last\_name`,`salary` from `employees`**

**order by `employee\_id`;**

### Example

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

## 4. 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.**

**select `first\_name`,`middle\_name`,`last\_name` from `employees`**

**order by `employee\_id`;**

### Example

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

## 5. 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**.

**select CONCAT(`first\_name`,'.',`last\_name`,'@softuni.bg') from `employees`;**

### Example

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

## 6. Find All Different Employee's Salaries

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

**select DISTINCT `salary` from `employees`**

**order by `salary` ASC;**

### Example

|  |
| --- |
| **Salary** |
| 12500.00 |
| 13500.00 |
| 43300.00 |
| … |

## 7. 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**.

**select \* from `employees` where `job\_title`='Sales representative'**

**order by `employee\_id`;**

### Example

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **id** | **First**  **Name** | **Last**  **Name** | **Middle Name** | **Job Title** | **Dept**  **ID** | **Mngr**  **ID** | **Hire Date** | **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 |
| … | … | … | … | … | … | … | … | … | … |

## 8. 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**.

**select `first\_name`,`last\_name`,`job\_title` from `employees` where `salary`>=20000 and `salary`<=30000**

**order by `employee\_id`;**

### Example

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

## 9. 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**.

**select CONCAT(`first\_name`,' ',`middle\_name`,' ',`last\_name`) from `employees` where `salary`=25000 or `salary`=14000 or `salary`=12500 or `salary`=23600**

**order by `employee\_id`;**

### Example

|  |
| --- |
| **Full Name** |
| Guy R Gilbert |
| Thierry B D'Hers |
| JoLynn M Dobney |

## 10. 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**.

**select `first\_name`,`last\_name` from `employees` where `manager\_id` is null**

**order by `employee\_id`;**

### Example

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

## 11. 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**.

**select `first\_name`,`last\_name`,`salary` from `employees` where `salary`>50000**

**order by `salary` desc;**

### Example

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

## 12. 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**.

**select `first\_name`,`last\_name` from `employees`**

**order by `salary` desc**

**LIMIT 5;**

### Example

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

## 13. 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.**

**select `first\_name`,`last\_name` from `employees` where `department\_id`>4 or `department\_id`<4;**

### Example

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

## 14. 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.**

**select \* from `employees`**

**order by salary DESC,first\_name,last\_name DESC,middle\_name;**

### Example

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **id** | **First**  **Name** | **Last**  **Name** | **Middle Name** | **job\_title** | **Dept**  **ID** | **Mngr**  **ID** | **Hire Date** | **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 |
| … | … | … | … | … | … | … | … | … | … |

## 15. Create View Employees with Salaries

Write a SQL query to create a view **v\_employees\_salaries** with **first name**, **last name** and **salary** for each employee. Submit your query statements as **Run skeleton, run queries & check DB.**

**create view v\_employees\_salaries as**

**select first\_name, last\_name, salary from employees;**

### Example

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

## 16. Create View Employees with Job Titles

Write a SQL query to create view **v\_employees\_job\_titles** with **full employee name** and **job title**. When middle name is **NULL** replace it with **empty string ('')**. Submit your query statements as **Run skeleton, run queries & check DB.**

**CREATE VIEW `v\_employees\_job\_titles` AS**

**SELECT**

**CONCAT(`first\_name`,**

**' ',**

**IFNULL(CONCAT(`middle\_name`, ' '), ''),**

**`last\_name`) AS 'full\_name',**

**`job\_title`**

**FROM**

**`employees`;**

### Example

|  |  |
| --- | --- |
| **full\_name** | **job\_title** |
| Guy R Gilbert | Production Technician |
| Kevin F Brown | Marketing Assistant |
| Roberto Tamburello | Engineering Manager |
| … | … |

## 17. Distinct Job Titles

Write a SQL query to find **all distinct job titles**. **Sort the result by job title alphabetically.** Submit your query statements as **Prepare DB & run queries.**

**select distinct job\_title from employees**

**order by job\_title;**

### Example

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

## 18. 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.**

**select \* from projects**

**order by start\_date,name**

**limit 10;**

### 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 |
| … | … | … | … | … |

## 19. 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.**

**select first\_name,last\_name,hire\_date from employees**

**order by hire\_date desc**

**limit 7;**

### 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 |
| … | … | … |

## 20. 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.**

**Update employees**

**set salary=salary\*1.12 where department\_id=1 or department\_id=2 or department\_id=4 or department\_id=11;**

**select salary from employees;**

### Example

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

# Part II – Queries for Geography Database

## 21. All Mountain Peaks

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

**select peak\_name from peaks**

**order by peak\_name;**

### Example

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

## 22. 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.**

**select country\_name,population from countries where continent\_code='EU'**

**order by population desc,country\_name**

**limit 30;**

### Example

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

## 23. 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.**

**SELECT country\_name, country\_code,**

**CASE currency\_code**

**WHEN 'EUR' THEN 'Euro'**

**ELSE 'Not Euro'**

**END AS currency FROM countries**

**ORDER BY country\_name;**

### Example

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

# Part III – Queries for Diablo Database

## 24. All Diablo Characters

Display the **name** of all **characters** in **alphabetical order.** Submit your query statements as **Prepare DB & run queries.**

**select name from characters**

**order by name;**

### Example

|  |
| --- |
| **name** |
| Amazon |
| Assassin |
| Barbarian |
| … |