# Упражнение: Съединяване на таблици

Тествайте решенията си в **Judge системата**: <https://judge.softuni.org/Contests/Practice/Index/4608#0>

# Част 1 – Заявки към базата данни SoftUni

## Адрес на работник

Напишете **SQL заявка** която да извлече:

* **EmployeeId**
* **JobTitle**
* **AddressId**
* **AddressText**

Върнете **първите 5** реда, сортирани по **AddressId** във **възходящ ред**.

### Пример

|  |  |  |  |
| --- | --- | --- | --- |
| **EmployeeId** | **JobTitle** | **AddressId** | **AddressText** |
| 142 | Production Technician | 1 | 108 Lakeside Court |
| 30 | Human Resources Manager | 2 | 1341 Prospect St |
| … | … | … | … |

## Отдели на служители

Напишете **SQL заявка** която да извлече:

* **EmployeeID**
* **FirstName**
* **Salary**
* **DepartmentName**

Филтрирайте само служители със **заплата**, **по-висока** от **15000**. Върнете **първите** **5** реда, **подредени** по **DepartmentID** във **възходящ ред**.

### Пример

|  |  |  |  |
| --- | --- | --- | --- |
| **EmployeeID** | **FirstName** | **Salary** | **DepartmentName** |
| 3 | Roberto | 43300.00 | Engineering |
| 9 | Gail | 32700.00 | Engineering |
| … | … | … | … |

## Служители без проект

Напишете **SQL заявка**, която да извлече:

* **EmployeeID**
* **FirstName**

Филтрирайте само служители **без проект**. Върнете **първите 3** реда, сортирани по **EmployeeID** във **възходящ** **ред**.

### Пример:

|  |  |
| --- | --- |
| **EmployeeID** | **FirstName** |
| 2 | Kevin |
| 6 | David |
| … | … |

## Служители с проект

Напишете **SQL заявка** която да извлече:

* **EmployeeID**
* **FirstName**

Филтрирайте само служители **с проект**, който **е започнал след** ‘**13.08.2002**’ и все още продължава (**няма крайна** дата). Върнете **първите 5** реда, сортирани по **EmployeeID** във **възходящ ред**.

### Пример

|  |  |  |
| --- | --- | --- |
| **EmployeeID** | **FirstName** | **ProjectName** |
| 1 | Guy | Racing Socks |
| 1 | Guy | Road Bottle Cage |
| … | … | … |

## Мениджър на служител

Напишете **SQL заявка** която да извлече:

* **EmployeeID**
* **FirstName**
* **ManagerID**
* **ManagerName**

Филтрирайте всички служители с **мениджър**, който има **ID**, равно на **3** или **7**. Върнете **всички** редове, сортирани по **EmployeeID** във **възходящ ред**.

### Пример

|  |  |  |  |
| --- | --- | --- | --- |
| **EmployeeID** | **FirstName** | **ManagerID** | **ManagerName** |
| 4 | Rob | 3 | Roberto |
| 9 | Gail | 3 | Roberto |
| … | … | … | … |

# Част 2 – Заявки към базата данни Geography

## Най-високите върхове в България

Напишете **SQL заявка** която да извлече:

* **CountryCode**
* **MountainRange**
* **PeakName**
* **Elevation**

Филтрира всички върхове в България с **надморска височина** над **2835 м**. Върнете **всички** редове, сортирани по **надморска** **височина** в **низходящ** **ред**.

### Пример

|  |  |  |  |
| --- | --- | --- | --- |
| **CountryCode** | **MountainRange** | **PeakName** | **Elevation** |
| BG | Rila | Musala | 2925 |
| BG | Pirin | Vihren | 2914 |
| … | … | … | … |

## Държави с реки

Напишете **SQL заявка** която да извлече:

* **CountryName**
* **RiverName**

Намерете **първите** **5** страни **с** или **без** реки в **Африка**. Сортирайте ги по **име** **на** **държава** във **възходящ ред**.

### Пример

|  |  |
| --- | --- |
| **CountryName** | **RiverName** |
| Algeria | Niger |
| Angola | Congo |
| Benin | Niger |
| Botswana | NULL |
| Burkina Faso | Niger |