# Lab: Databases and SQL Language Intro

Problems for exercises and homework for the "Databases and SQL" course from the official "Applied Programmer" curriculum.

You can check your solutions here: <https://judge.softuni.org/Contests/3369/Databases-and-SQL-Language-Intro-Lab>

**Note**: if you don't have installed **MS** **SQL** **Server** **LocalDB** and **SQL** **Server** **Management** **Studio**, follow the installation guide **01. Databases-and-SQL-Server-Installation-Guide.docx** to install:

* SQL Server LocalDB – this is the database engine
* SQL Server Management Studio (SSMS) – this is the tool to browse the database

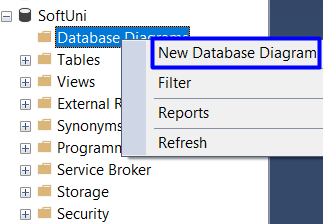
## Import the Database

**Unzip** the **01.Databases-and-SQL-Language-Lab.zip** file, containing the databases. Start your **SQL Server Manager Studio**.

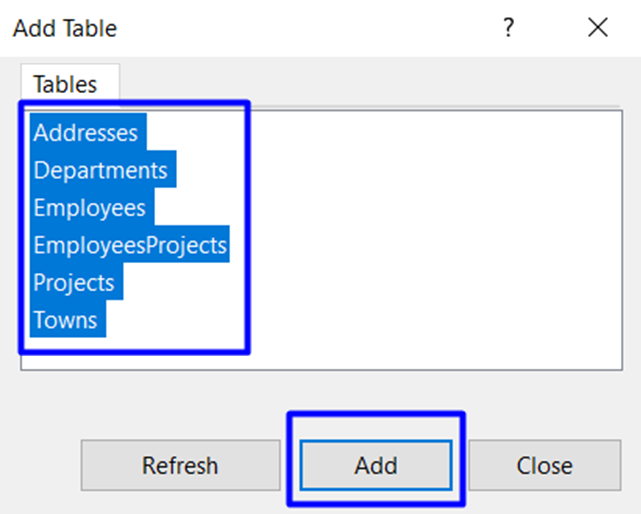
Import the **SoftUni database**. Follow the steps from **01.Import-the-DB-Guide.docx**, if you already have this database, delete it and import it again.

## Create E/R Diagram

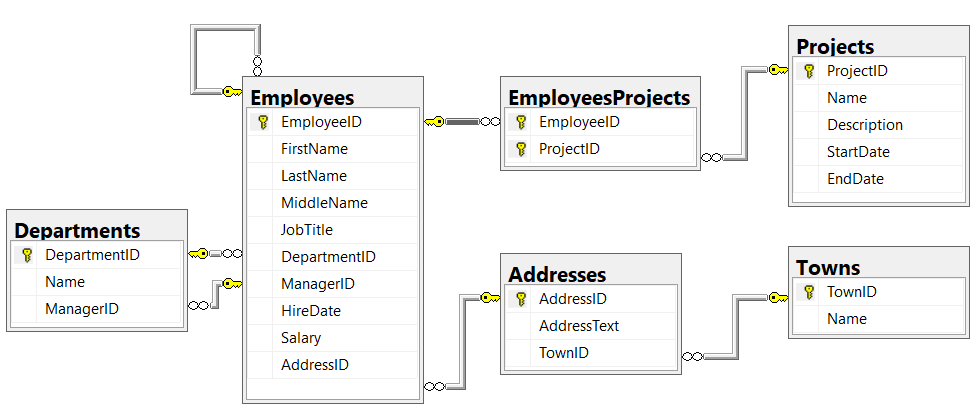
In order to create an **E/R diagram** of the SoftUni database, go to Object Explorer -> Databases -> SoftUni and **right-click** on Database Diagrams. Choose [New Database Diagram].



Then, **select all tables** and press [Add].



When tables are **loaded**, press [Close] and your diagram is already created. You can always **re-arrange** table blocks. The diagram should be something like this:



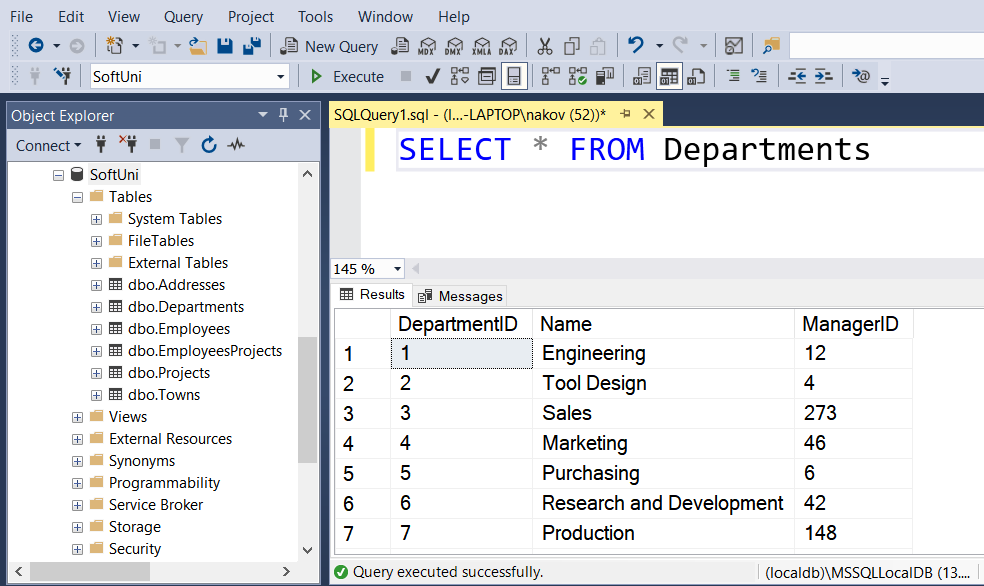
## Display All Information about the Departments

Write a SQL query to find **all available information about the Departments**.

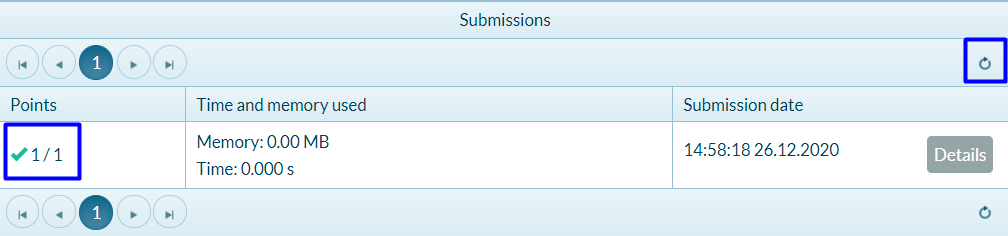
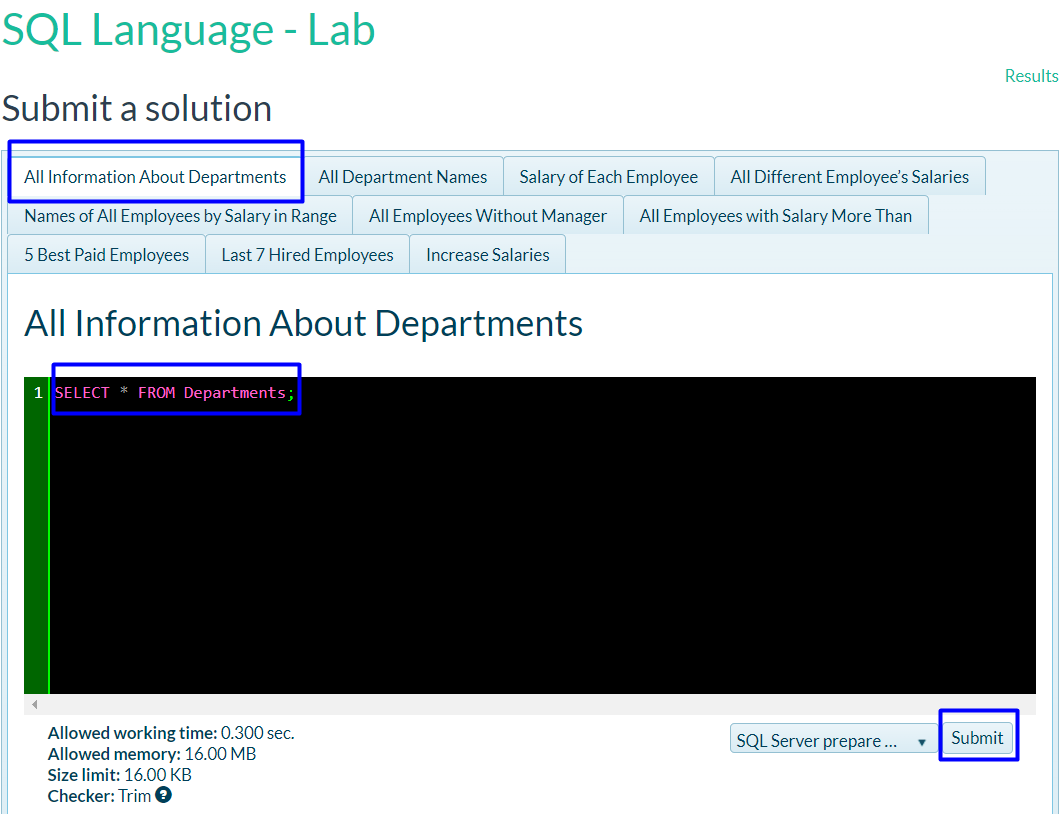
### Example

|  |  |  |
| --- | --- | --- |
| **DepartmentID** | **Name** | **ManagerID** |
| 1 | Engineering | 12 |
| 2 | Tool Design | 4 |
| 3 | Sales | 273 |
| … | … | … |

### Hints



### Submit Your Solution in Judge



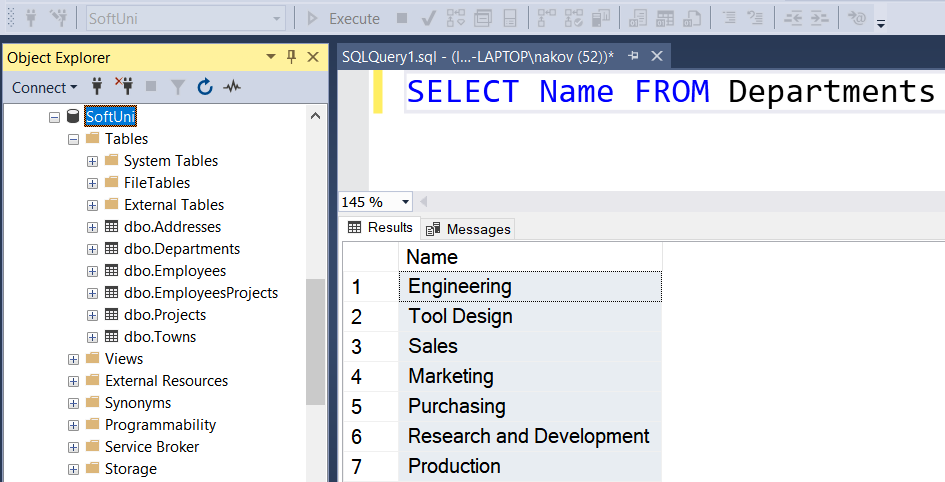
## Display All Department Names

Write SQL query to find **all Department names**.

### Example

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

### Hints



## Salary of Each Employee

Write SQL query to find the **first name**, **last name** and **salary** of each employee.

### Example

|  |  |  |
| --- | --- | --- |
| **FirstName** | **LastName** | **Salary** |
| Guy | Gilbert | 12500.00 |
| Kevin | Brown | 13500.00 |
| Roberto | Tamburello | 43300.00 |
| … | … | … |

## All Different Employee’s Salaries

Write a SQL query to find **all different employee’s salaries**. Show only the salaries.

### Example

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

## 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].**

### Example

|  |  |  |
| --- | --- | --- |
| **FirstName** | **LastName** | **JobTitle** |
| Rob | Walters | Senior Tool Designer |
| Thierry | D'Hers | Tool Designer |
| JoLynn | Dobney | Production Supervisor |
| … | … | … |

## All Employees Without Manager

Write a SQL query to find **first and last names** about those employees that **does not have a manager**.

### Example

|  |  |
| --- | --- |
| **FirstName** | **LastName** |
| Ken | Sanchez |
| Svetlin | Nakov |
| … | … |

## 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.

### Example

|  |  |  |
| --- | --- | --- |
| **FirstName** | **LastName** | **Salary** |
| Ken | Sanchez | 125500.00 |
| James | Hamilton | 84100.00 |
| … | … | … |

## 5 Best Paid Employees.

Write SQL query to find **first and last names** about **5 best paid Employees** ordered **descending by their salary.**

### Example

|  |  |
| --- | --- |
| **FirstName** | **LastName** |
| Ken | Sanchez |
| James | Hamilton |
| … | … |

## Last 7 Hired Employees

Write a SQL query to find **last 7 hired employees**. Select **their first, last name and their hire date**.

### Example

|  |  |  |
| --- | --- | --- |
| **FirstName** | **LastName** | **HireDate** |
| Rachel | Valdez | 2005-07-01 00:00:00 |
| Lynn | Tsoflias | 2005-07-01 00:00:00 |
| Syed | Abbas | 2005-04-15 00:00:00 |
| … | … | … |

## 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.

### Example

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

## Employee Address

Write a query that selects:

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

Return the **first 5** rows **sorted** by **AddressId** in **ascending** order.

### Example

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

## Addresses with Towns

Write a query that selects:

* **FirstName**
* **LastName**
* **Town**
* **AddressText**

**Sorted** by **FirstName** in **ascending** order then by **LastName**. Select **first 50** employees.

### Example

|  |  |  |  |
| --- | --- | --- | --- |
| **FirstName** | **LastName** | **Town** | **AddressText** |
| A.Scott | Wright | Newport Hills | 1400 Gate Drive |
| Alan | Brewer | Kenmore | 8192 Seagull Court |
| … | … | … | … |

## Sales Employee

Write a query that selects:

* **EmployeeID**
* **FirstName**
* **LastName**
* **DepartmentName**

**Sorted** by **EmployeeID** in **ascending** order. Select only **employees** from “**Sales**” department.

### Example

|  |  |  |  |
| --- | --- | --- | --- |
| **EmployeeID** | **FirstName** | **LastName** | **DepartmentName** |
| 268 | Stephen | Jiang | Sales |
| 273 | Brian | Welcker | Sales |
| … | … | … | … |

## Employee Departments

Write a query that selects:

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

Filter only **employees** with **salary higher than 15000**. Return the **first 5** rows **sorted** by **DepartmentID** in **ascending** order.

### Example

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

## Employees Without Project

Write a query that selects:

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

Filter only **employees** **without** a **project**. Return the **first 3** rows **sorted** by **EmployeeID** in **ascending** order.

### Example

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

## Employees Hired After

Write a query that selects:

* **FirstName**
* **LastName**
* **HireDate**
* **DeptName**

Filter only **employees** **hired after 1.1.1999** and are from either **"Sales"** or **"Finance"** departments, s**orted** by **HireDate** (**ascending**).

### Example

|  |  |  |  |
| --- | --- | --- | --- |
| **FirstName** | **LastName** | **HireDate** | **DeptName** |
| Debora | Poe | 2001-01-19 00:00:00 | Finance |
| Wendy | Kahn | 2001-01-26 00:00:00 | Finance |
| … | … | … | … |