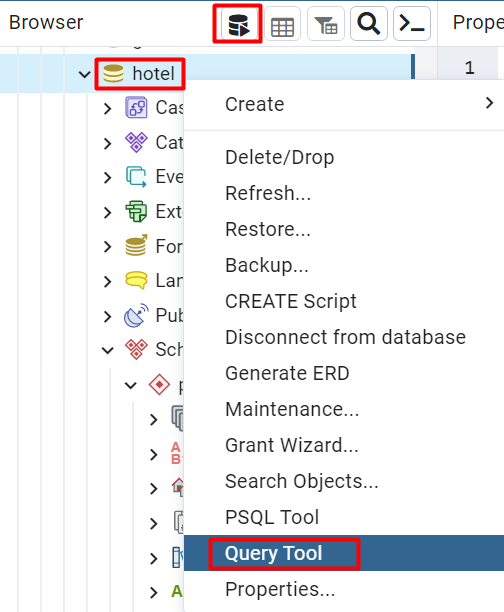
# Lab: Basic CRUD

This document defines the **lab exercise assignments** for the [**PostgreSQL course @ Software University**](https://softuni.bg/modules/137/python-db).

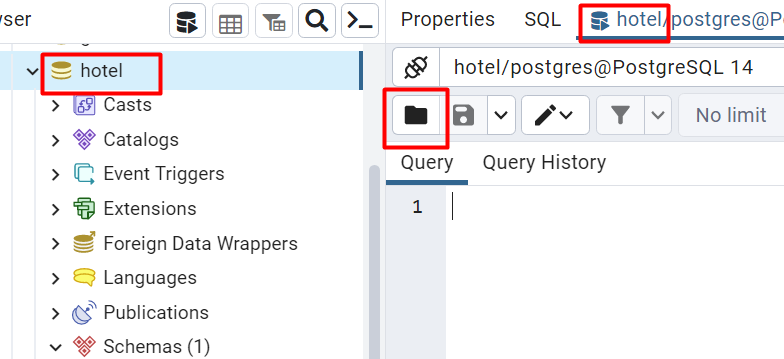
## Initial Steps:

Download the file **hotel\_db.sql** fromyourResources section.

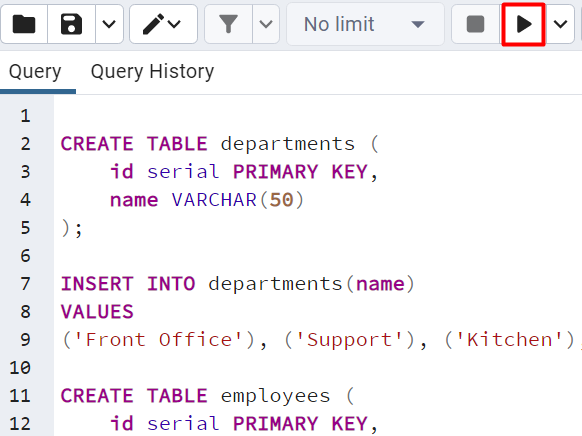
Create a database "**hotel**" and open its **Query Tool**. You can use either the **Query icon** or **Query Tool** from the drop-down menu. Make sure you have selected your database **hotel.**



Import the file by clicking on the **Folder** icon.



Navigate to file **hotel\_db.sql** and select it. Queries will be loaded. Press the **Play** icon to execute them. This way you will create all tables and populate them with bulk data.



Get familiar with the **hotel** database and its **tables**. You will use them in the following exercises to make queries and retrieve data.

**Exercises from the following section should be submitted to** [**Judge Contest**](https://judge.softuni.org/Contests/4102/Basic-CRUD-Lab) **(tasks 1 to 7).**

### **Select and Display Employee Information by Concatenating Columns**

Write a query to select all employees and retrieve information about their **id**, **first\_name**, **last\_name (as Full Name)**,and **job\_title (as Job Title)**.

#### **Example**

|  |  |  |
| --- | --- | --- |
| **id** | **Full Name** | **Job Title** |
| 1 | John Smith | Manager |
| 2 | John Johnson | Customer Service |
| 3 | Smith Johnson | Porter |
| … | … | … |

### **Select Employees by Filtering and Ordering**

Write a query to select all employees (**id, first\_name, and last\_name (as full\_name), job\_title, salary**) whose salaries are **higher than 1000.00**, **ordered by id.** Concatenate fields **first\_name** and **last\_name** into '**full\_name**'.

#### **Example**

|  |  |  |  |
| --- | --- | --- | --- |
| **id** | **full\_name** | **job\_title** | **salary** |
| 3 | Smith Johnson | Porter | 1100 |
| 4 | Peter Petrov | Front Desk Clerk | 1100 |
| 5 | Peter Ivanov | Sales | 1500.23 |
| … | … | … | … |

### **Select Employees by Multiple Filters**

Write a query to retrieve information about employees, who are in **department 4** and have a salary **higher or equal to 1000**. Order the information by **id**.

#### **Example**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **id** | **first\_name** | **last\_name** | **job\_title** | **department\_id** | **salary** |
| 3 | Smith | Johnson | Porter | 4 | 1100 |
| 9 | Nikolay | Ivanov | Housekeeping | 4 | 1600 |

### **Insert Data into Employees Table**

Insert new records into table **employees** by writing a query. Select all employees’ info to check the new entries.

**New values:**

**Samantha Young, Housekeeping, 4, 900**

**Roger Palmer, Waiter, 3, 928.33**

Submit **both** **INSERT** and **SELECT** queries.

#### **Example**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **id** | **first\_name** | **last\_name** | **job\_title** | **department\_id** | **salary** |
| … | … | … | … | … | … |
| 10 | Samantha | Young | Housekeeping | 4 | 900 |
| 11 | Roger | Palmer | Waiter | 3 | 928.33 |

### **Update Employees Salary**

Update all employees' salaries whose **job\_title** is "**Manager**" by **adding 100**.

**Retrieve** information from table **employees** for all **managers**.

Submit **both** **UPDATE** and **SELECT** queries.

#### **Example**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **id** | **first\_name** | **last\_name** | **job\_title** | **department\_id** | **salary** |
| 1 | John | Smith | Manager | 1 | 1000 |

### **Delete from Table**

Write a query to delete all employees from the **employees** table who are in department **2 or 1**. Then **select** all from table **employees** and order the information by **id**.

Submit **both** **DELETE** and **SELECT** queries.

#### **Example**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **id** | **first\_name** | **last\_name** | **job\_title** | **department\_id** | **salary** |
| 3 | Smith | Johnson | Porter | 4 | 1100 |
| 6 | Ivan | Petrov | Waiter | 3 | 990 |
| 7 | Jack | Jackson | Executive Chef | 3 | 1800 |
| 9 | Nikolay | Ivanov | Housekeeping | 4 | 1600 |

### **Create a View for Top Paid Employee**

Write a query to create a **view** that selects all information about the top-paid employee from the **employees** table in the **hotel** database. Call your view to check the results.

Submit **both** **CREATE VIEW** and **SELECT** queries.

#### **Example**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **id** | **first\_name** | **last\_name** | **job\_title** | **department\_id** | **salary** |
| 8 | Pedro | Petrov | Front Desk Supervisor | 1 | 2100 |