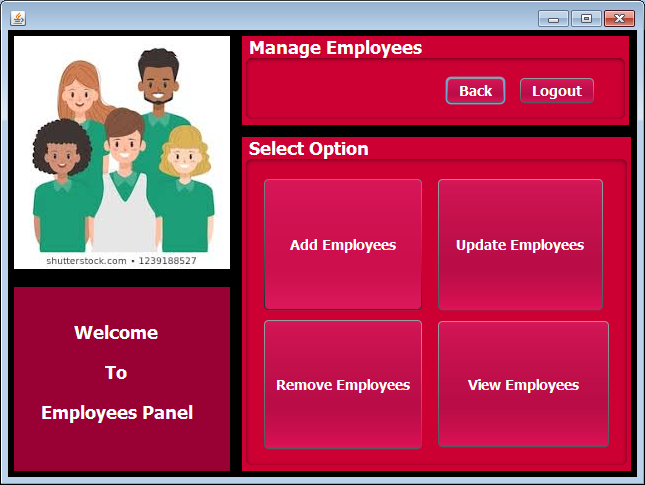
**Designing The ManageEmplyeesFrame**

****

**STEPS TO BE DONE IN** **ManageEmplyeesFrame**

In the **ManageEmplyeesFrame** we need to do following steps:

1. Allow the user to **logout**
2. Write code for the "**Add Employees**" Button . When this button is clicked it should:
   1. Generate an **Employee-Id**
   2. Display it in the **Employee-id** field.
3. When the "**Back**" button is clicked we must go back to the **AdminOptionsFrame**

**Displaying The Username**

Same as **AdminOptionsFrame**

**Allowing The User To Logout**

Same as **AdminOptionsFrame**

**THE TABLE USED IN ManageEmplyeesFrame**

**EMPLOYEES:**

|  |  |  |
| --- | --- | --- |
| **Column Name** | **Data Type** | **Description** |
| **EMPID** | **Varchar2(10)** | **Contains Userid** |
| **EMPNAME** | **Varchar2(10)** | **Contains password** |
| **JOB** | **Varchar2(10)** | **Contains job profile** |
| **SALARY** | **NUMBER(6,2)** | **Contains Salary** |

**WRITING THE CODE FOR BUTTON IN AddEmplyeesFrame**

**Generating The Employee-Id**

To generate the **Employee-Id** , we use a simple logic:

**a. Find out number of rows in the EMPLOYEE Table**

**b. Add 1 to rowcount and prefix it with "E" which is the new Employee-Id. For Employeeple if total rows in the EMPLOYEE Table is 3 then next Employee-Id will be "E104"**

**How To Find Number Of Rows**

To find number of rows in the **EMPLOYEE** Table we do the following:

a. Create the **EmployeeDAO** class with a method called **getEmployeeId( )**.

b. This method should execute the following SQL query:

***Select count(\*) as totalrows from Employee***

and return value of **count(\*) +1** as the **Employee-Id** prefixed with "**E**"

c. So the prototype of the method is:

***public static String getEmployeeId()throws SQLException***