Display menu to the HR should be as follows

* 1. Enter Employee Details
  2. Exit

1. **Enter Employee Details**

When the HR selects this option, it allow the HR to enter the employee details as shown in the below sample screen. When all the details entered are added successfully, it should print the message “**Employee has been successfully registered along with the <Employee ID>**”.

**Note**:

* + 1. Employee ID should be generated using Random
    2. Employee Travel Allowance should be calculated according to the role provided in the below table
    3. Validation should be taken care for all input fields

|  |  |
| --- | --- |
| **Role** | **TA ( % of basic salary)** |
| Senior Manager | 25 |
| Manager | 20 |
| Senior Consultant | 15 |
| Consultant | 10 |

**Sample Screen**

**Enter the name of the Employee** : rakesh

**Enter Employee phone number**: 9768587350

**Employee Role**: Manager

**Salary: 50000**

**Mail Id**: rakesh@capgemini.com.

The below message should be generated

**Employee has been successfully Register along with the 10001**

1. Exit

**Classes to be created**

**com.capgemini.emp.ui**

Class Client {

public static void main (

// User Interface which display the menu and accept the Input from user

// Create object for service and execute the respective methods}

**com.capgemini.emp.bean**

Class Employee { // Bean class for Employee with all required properties}

**com.capgemini.emp.service** // Service utility classes and Interfaces

Interface IEmployeeService {

int addEmployee (Employee emp);

Employee getEmployee (int empId);

}

Class EmployeeService implements IEmployeeService {

int addEmployee (Employee emp){ . . . . . }

Employee getEmployee (int empId) { . . . . . }

}

**com.capgemini.emp.dao //** Class to perform Data Accessing logic

Interface IEmployeeDao {

int addEmployee (Employee emp);

Employee getEmployee (int empId);

}

Class EmployeeDao implements IEmployeeDao{

int addEmployee (Employee emp){ . . . . . }

Employee getEmployee (int empId) { . . . . . }

}

Add appropriate user defined exception classes and any other supporting classes required.