**Create a new project with three classes**

1. **SingleAddress**
2. **AddressBook Interface**
3. **AddressBookImpl Class**
4. **TestAddressBook**

Put these classes in an appropriately-named projects & namespaces other than the default names.

The class **SingleAddress** should be able to store the following information:

* last name
* first name
* street address
* city
* state
* country
* postal code.

Create getter and setter methods as necessary. The class **AddressBookImpl** needs to store a collection of addresses. It needs to have methods to do the following:

1. Add a new address to the address book.

Handle exception, when adding the address of same person again. Consider first name to check the duplicate. Hint: Create a user defined exception called “**DuplicateAddressException**”

1. Remove an address from the address book using first name field.
2. Write a Method to display all the address’s available in the AddressBook.
3. Override the “toString” method to return a string as below
   * First name, last name, street address, city, state.

Finally, the **TestAddressBook** class will test the functionality above.

**Note: Please follow the coding standards and best practices**

1. **Naming Conventions [ Class, Method, Variable, Package]**
2. **Best Practices to be followed in Core Java**

**Turn to next page.**

1. **Example Tables:**

**Table: DEPARTMENT**

|  |  |  |
| --- | --- | --- |
| **Column Name** | **Data Type** | **Constraint** |
| DNO | NUMBER(2) | Primary Key |
| DNAME | VARCHAR2(10) | Unique |
| LOCATION | VARCHAR2(10) | Not Null |

**Table: EMPLOYEE**

|  |  |  |
| --- | --- | --- |
| **Column Name** | **Data Type** | **Constraint** |
| EMPNO | NUMBER(2) | Primary Key |
| ENAME | VARCHAR2(10) | Unique |
| MGR | VARCHAR2(10) | Not Null |
| HIREDATE | DATE | >=SYSDATE |
| DNO | NUMBER(2) | Foreign Key |
| SAL | NUMBER(7,2) | >500 |
| COMM | NUMBER(7,2) |  |
| DNO | NUMBER(2) | Foreign Key |

**List the department names with location details whose number of employees are more than 4 (Descriptive Question)**

1. **List out the employee’s who are all earning more than their respective boss?**
2. SELECT e.empno, e.ename, e.sal FROM employees e JOIN emp b ON (e.mgr=b.empno) AND e.sal>b.sal;
3. SELECT e.empno, e.ename, e.sal FROM employees e JOIN emp b ON (b.mgr=e.empno) AND e.sal>b.sal;
4. SELECT e.empno, e.ename, e.sal FROM employees e JOIN emp b ON (e.mgr=b.empno) AND e.sal<b.sal;
5. SELECT e.empno, e.ename, e.sal FROM employees e INNER JOIN emp b USING (e.mgr=b.empno) AND e.sal>b.sal;

**@@@@@@@@@@@@@@ All the Best Happy Learning @@@@@@@@@@@@@@@**