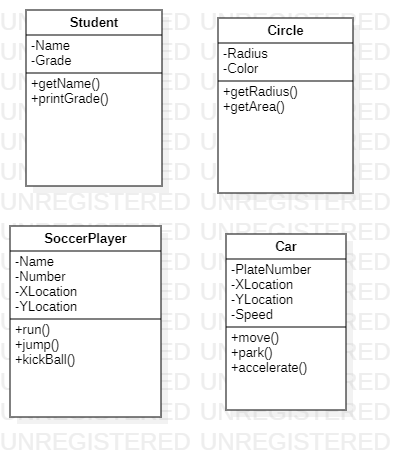
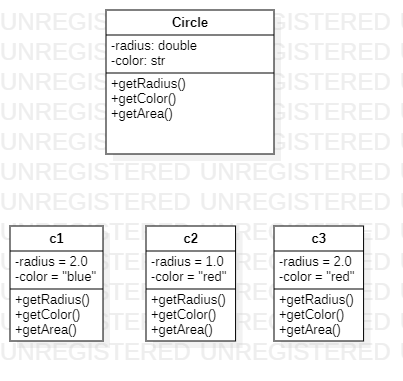
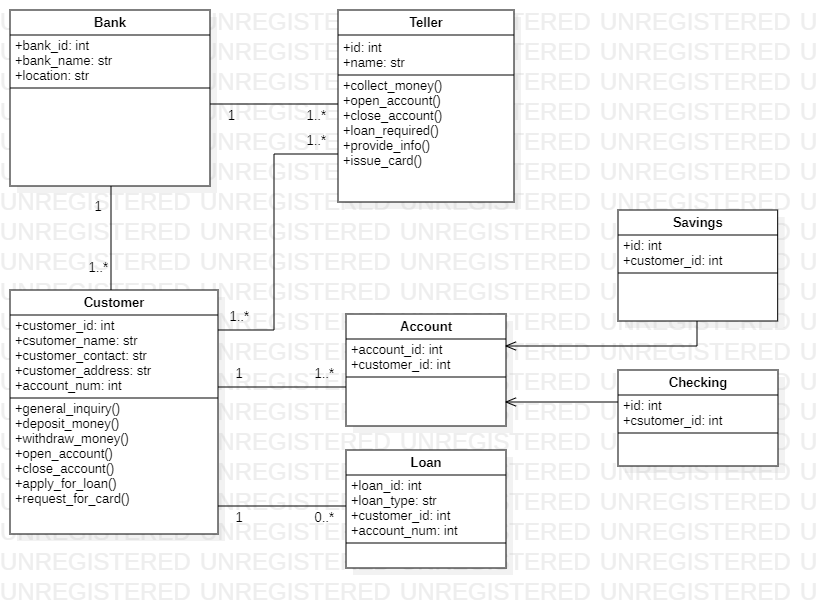
* **Exercise 01:**

****

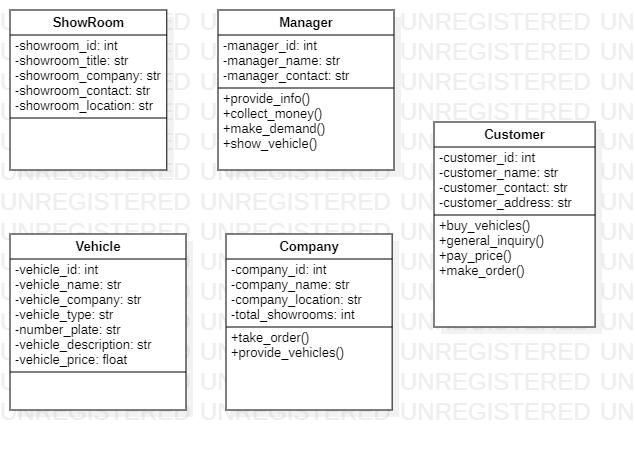
* **Exercise 02:**

****

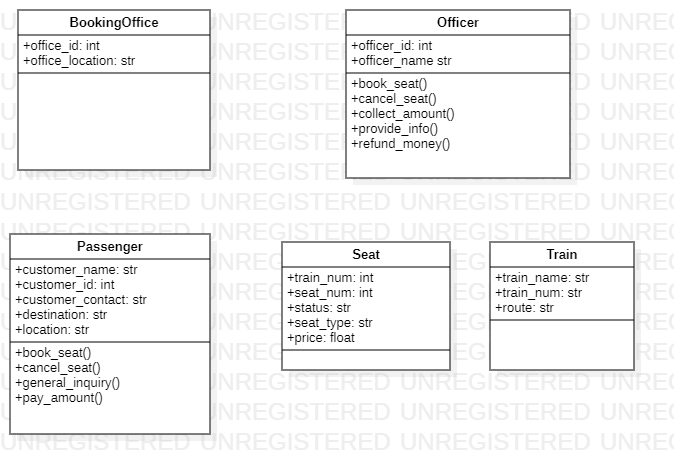
* **Exercise 03:**
* **Exercise 04:**

****

* **Exercise 05:**

****

* **Exercise 06:**

****

* **Exercise 07:**

1. **Class:** Class is defined as the template to create different objects. Class is like a blueprint in which details of object can be filled. It is the building of Object Oriented Programming. It makes code reusable, small and organized.

The real world example of class is like we have **Plants**. Plants is a class however there are many type of plants so they are objects of Plants class.

1. **Objects:** Objects can be defined as the collection of data and different methods is called object. Python is an Object Oriented Language. Objects are created after the definition of class.

The real world example of objects is like there are many kind of animals so Animal can be treated as class and kinds of animals can be treated as objects.

1. **Instances:** Instance is the synonym of Object. However, sometimes objects are also called instance objects. The word instance emphasizes the distinct identity of an object.
2. **Association:** Association is defined as using relationship between two or more class of different objects. In UMl, three type of associations can be used between class diagrams. These types are as following:
3. **Simple Association**
4. **Aggregation**
5. **Composition**
6. **Access Modifiers:** The Access Modifiers are special type of prefixes that are used to limit the access of different variables. Access Modifiers are used to change the scope of the variables and methods.

There are three type of access modifiers which are listed as following with their prefixes with respect to Python:

1. **Public +**
2. **Private –**
3. **Protected #**

**The End**