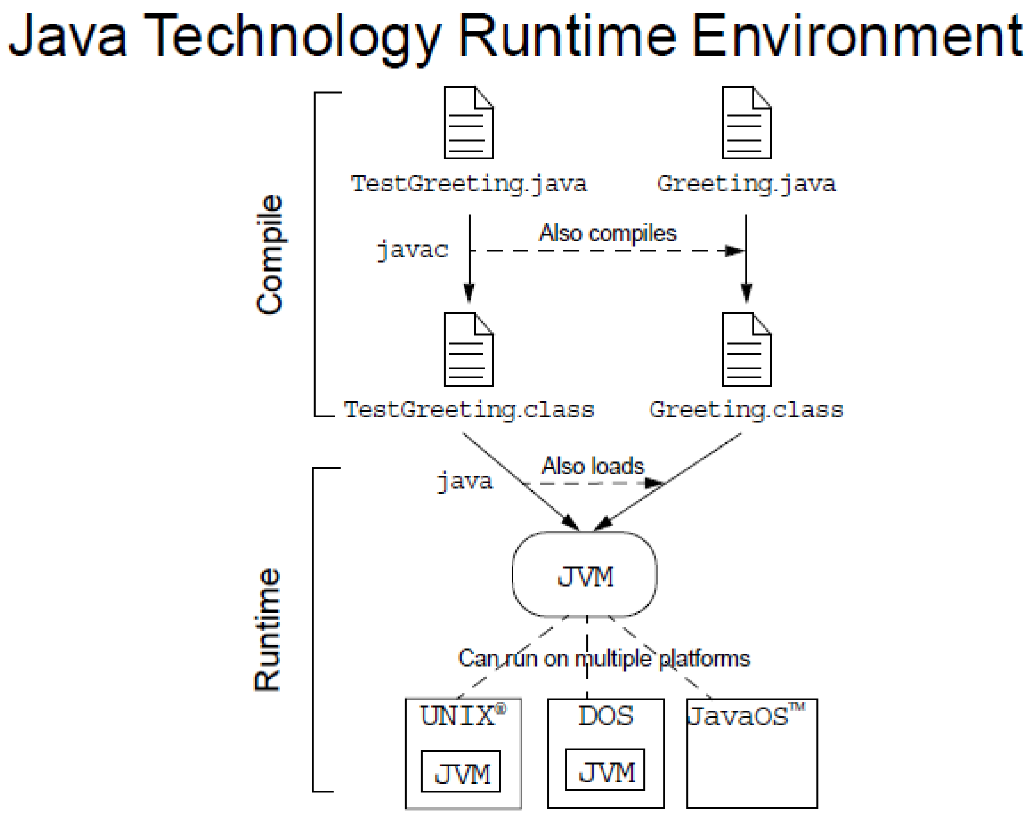
1. JVM Tasks

JVM performed three tasks, these are :

a. Load Code
Perform by the class loader which load all classes needed for the execution of a program. It adds more security characteristics in this phase. The memory layout of the executable file is determined.

b. The Byte Code Verifier: The JVM puts the code through a byte-code verifier that test a format of code fragmentation and checks code fragments for illegal code.

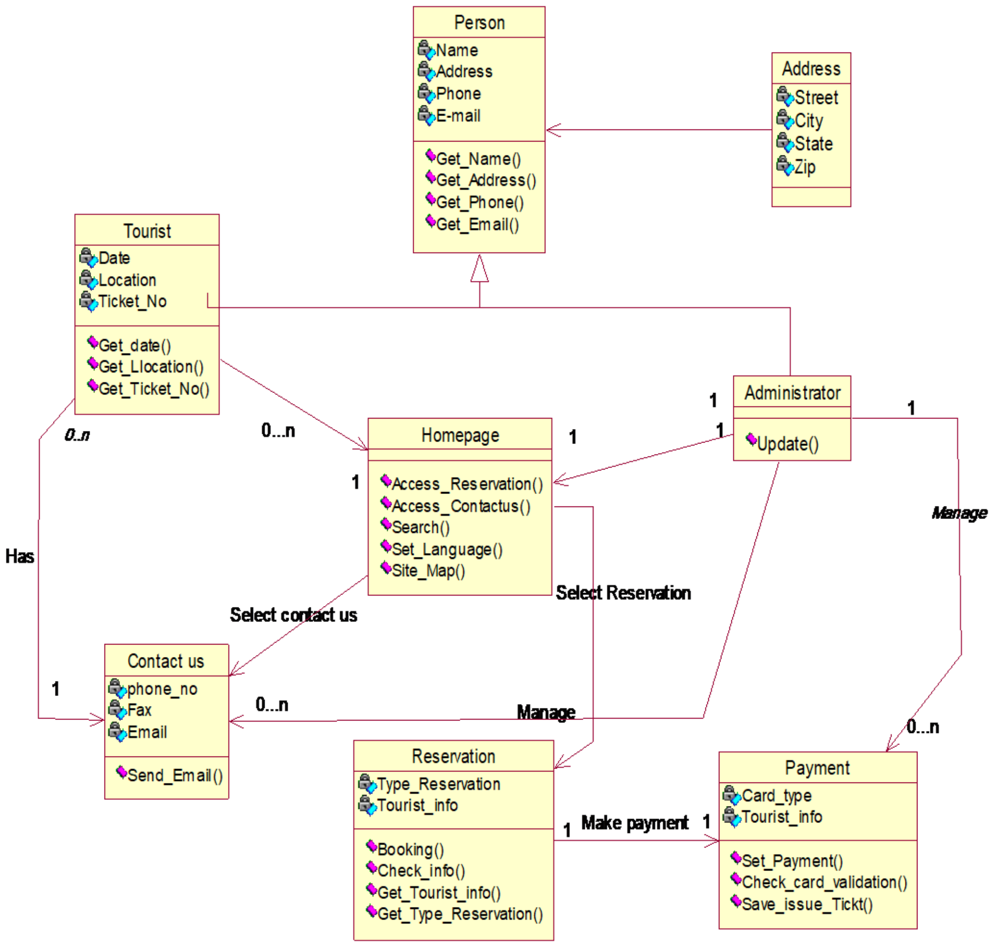
c. The Verification Process : it performs the following tasks.
- The classes adhere to the class file format of the JVM specifications
- There is no access restriction violations.
- The code ensures nor operand stack overflow or underflow.
- The types of parameters for all operational codes are correct.



Java Technology Runtime Environment

2. Object Oriented Programming

The Analysis and Design Phase :
Analysis describes what the system needs to do: Modeling the real-world, including actors and activities, objects, and behaviors.
Design describes how the system does it:
• Modeling the relationships and interactions between objects and actors in the system
Example: E-tourism Class Diagram



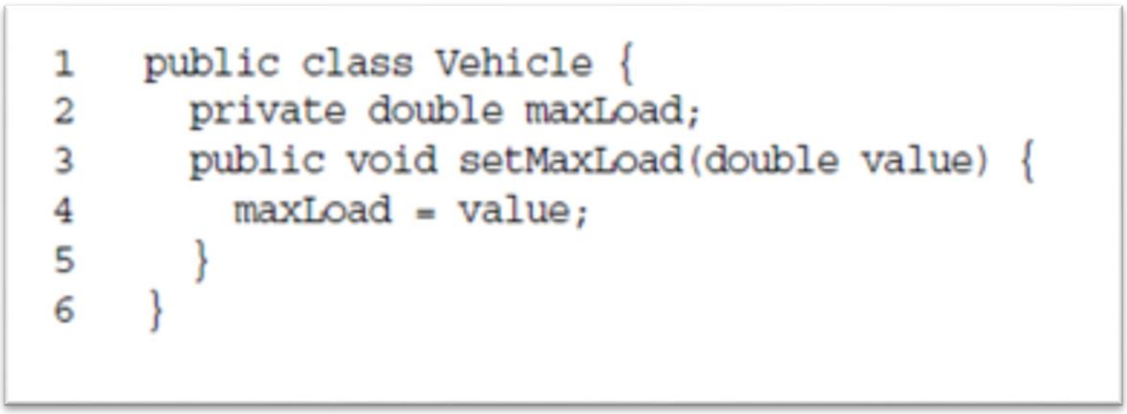
3. Declaring Java Technology Classes

a. The Basic syntax of a Java class:

<modifier>* class <class_name> {
<attribute_declaration>*
<constructor_declaration>*
<method_declaration>*

} end Class

Example
:



```
public class Vehicle {
  private double maxLoad;
  public void setMaxLoad(double value) {
    maxLoad = value;
  }
}
```

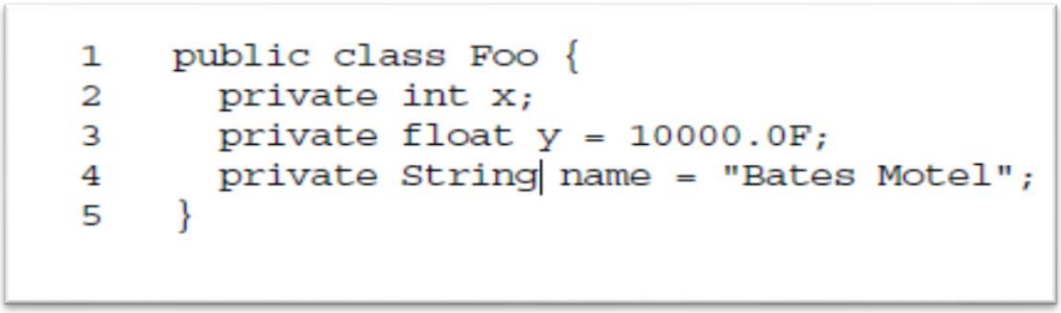
Question1: According to Syntax in above , specify class name and its modifier?

b. The Default Constructor

There is always at least one constructor in every class. If the writer does not supply any constructors, the default constructor is present automatically:
The default constructor takes no arguments and the default constructor body is empty.
The default enables you to create object instances with new Xxx()without having to write a constructor.

c. Basic syntax of an attribute:

<modifier>* <type> <name> [= <initial_value>];

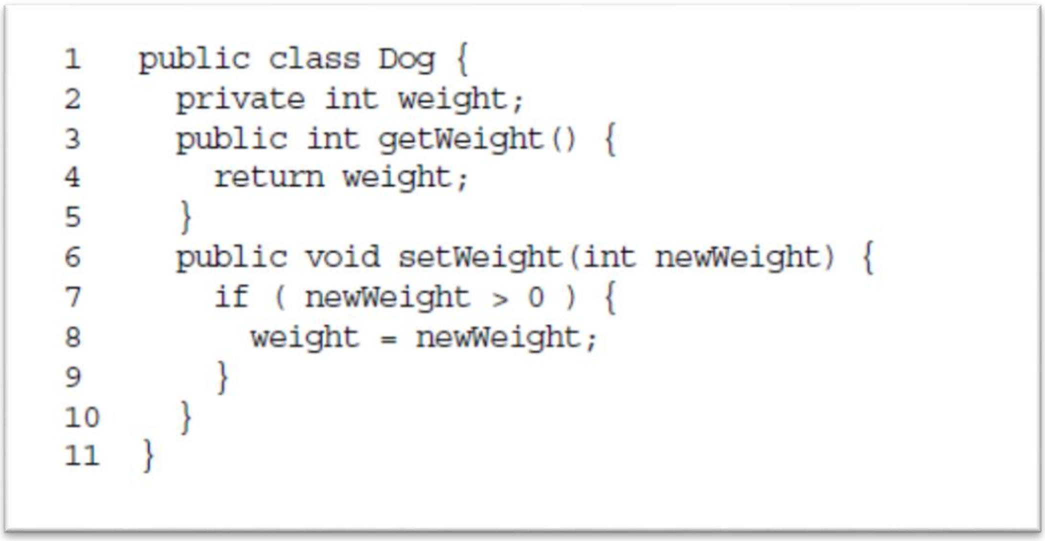


```
public class Foo {
  private int x;
  private float y = 10000.0F;
  private String name = "Bates Motel";
}
```

Question2: According to Syntax in above, specify class attribute(s) and variables

d. Declaring Methods

<modifier> <return_type> <name>* (*<argument>**)
{
*<statement>**
} end method



```
public class Dog {
  private int weight;
  public int getWeight() {
    return weight;
  }
  public void setWeight(int newWeight) {
    if ( newWeight > 0 ) {
      weight = newWeight;
    }
  }
}
```

e. Accessing Object Members

The *dot* notation is: ***<object>.<member>***

- This is used to access object members, including attributes and methods.

```
public static void main (String args []) {
        Dog d= new Dog ();
        d.setWeight(42);
        d.weight = 42; // only permissible if weight is public
  }
```

تعليق : ممكن ان نصل الى ال Class Dog من اي كلاس ثاني بشرط ان يكون الاول من نوع Public .

3. Access Control

Modifier	*Same Class*	*Same Package*	*SubClass*	*Universe*
private	Yes			
default	Yes	Yes		
protected	Yes	Yes	Yes	
public	Yes	Yes	Yes	Yes