

Experiment – 3

Class-Object and Methods

LEARNING OUTCOMES:

Student will be able to

1. Define class and object
2. Declare member functions and data members of a class
3. Access the data members and member functions by creating object of class

OBJECTIVE:

To Develop a Java program that Creates a Class, an Object and adding methods.

REQUIRED APPARATUS:

1. Notepad/Editors (VS Code, Atom, Brackets, Notepad++...), JDK 1.7 and above version.
2. Personal Computer with 2GB RAM, 320GB HDD and Pentium2 processor or above

PRECAUTIONS AND SAFETY MEASURES FOR A COMPUTER LAB

1. Don't touch the switch boards with wet hands.
2. Don't operate a system if the walls are wet.
3. Keep the food and beverages outside workspace
4. Shutdown and switch off properly the systems to avoid the system crash.
5. Keep footwear outside the lab so as to protect equipment from dust.
6. Know the place of fire extinguisher in lab

BRIEF THEORY

CLASS, OBJECTS AND member FUNCTION:

Class: Java is an Object-Oriented programming language. In Java, the classes and objects forms the basis for implementing any problem. Classes provide a means of bundling data and functionality together. The data of the class is called as data members and functions which access that data are called member functions of the class.

Syntax:

Class classname

{

Data member1, member2...;

```
Member function1()
Memberfunction2()
}
```

Object: An object is a variable of the class type, that accesses the data members and member functions of a class.

Syntax for declaring an object of a class:

classname object1=new classname(); // classname() it is a constructor

Method/ Function:

A method is a block of code that performs a specific task. In Java, there are two types of methods:

- **User-defined Methods:** We can create our own method based on our requirements.
- **Standard Library Methods:** These are built-in methods in Java that are available to use in programs.

Syntax for writing a member function:

```
Return type method_name()
{
    Body of statements;
}
```

PROCEDURE:

- Step1: Open the editor
- Step2: Type the program
- Step3: Define class by using class name Employee
- Step 4: Declare the necessary variables for class operations
- Step 5: Define a constructor by using class name and assign employee name
- Step 6: Initialize the code in function body for all the objects of a class
- Step 7: Save the file by using class name (Empolyee.java) with the extension as . java
- Step 8: Create another class with in the same file, by using the class name as EmpolyeeTest
- Step 9: Create required no of objects by using class name.
- Step 10: Create main method which is used provide static memory for performing object related operations
- Step11: Call the methods with the help of created objects
- Step 12: save and Compile the program using javac command in command prompt
- Step 13: Check and correct the errors then recompile
- Step 14: Run the program by the command java classname in the command prompt.
- Step 15: If the output corresponds to input then the program is successfully executed.

```

import java.io.*;

class Employee {
    int age;
    String name, designation;
    double salary;

    public Employee(String name) {
        this.name = name;
    }

    public void empAge(int empAge) {
        this.age = empAge;
    }

    public void empDesignation(String empDesig) {
        this.designation = empDesig;
    }

    public void empSalary(double sal) {
        this.salary = sal;
    }

    public void printEmployee() {
        System.out.println("Name: " + name);
        System.out.println("Age: " + age);
        System.out.println("Designation: " + designation);
        System.out.println("Salary: " + salary);
    }
}

public class EmployeeTest {
    public static void main(String args[]) {
        Employee empOne = new Employee("Dora");
        Employee empTwo = new Employee("Vani");

        empOne.empAge(30);
        empOne.empDesignation("Software Developer");
        empOne.empSalary(90000);
        empOne.printEmployee();

        empTwo.empAge(29);
        empTwo.empDesignation("Software Engineer");
        empTwo.empSalary(70000);
        empTwo.printEmployee();
    }
}

```

Note: we have to save this file by using class name only. Another important aspect is we need to run/Execute file by Using class in which main method is existing.

Out put:

C:\> javac Employee.java

C:\> javac EmployeeTest.java

C:\> java EmployeeTest

Name:Dora

Age:30

Designation: Software Developer

Salary:90000

Name:Vani

Age:29

Designation:Software Engineer

Salary:70000

Here, we have created a employee class, initialized the data member using constructor , methods are used to manipulate the data members . The methods and data members are accessed by creating object of the employee class.