PRACTICALS & ASSIGNMENTS
Object Oriented Programming with JAVA

Prof. Hardik P. Jagad

Subject Code: 2150704
List of Practical

Practical – 1 (Aim: To implement fundamental concept of JAVA)
1. Write a Java Program that displays “Hello World”.
2. Write a Java Program that will display Factorial of the given number (number given by command prompt).
3. Write a Java Program that will display 25 Prime nos.
4. Write a Java Program that will accept command-line arguments and display the same.
5. Write an application that creates a two dimension array with int values. The first, second and third elements should be arrays with one, two and three numbers respectively. Display the length of each dimension.
6. Write a program to convert rupees to dollar. 60 rupees = 1 dollar.
7. Write a program that calculate percentage marks of the student if marks of 6 subjects are given.
8. Write a program to enter two numbers and perform mathematical operations on them.

Practical – 2 (Aim: To implement concept of class and constructor)
1. Write an application that displays the use of at least five methods of math class.
2. Write an application that accepts one command-line argument. Convert this to an int and display a line reporting if that number is even or odd.
3. Write a program which receipts a character and nine numbers as command line arguments. if values of a character passed at the command line argument, is either a or A then program should sort the nine numbers in ascending order and if the value of a character happens to be either d or D then program should sort the nine numbers in descending order.
4. A marklist contains reg. no and marks for three subjects sub1, sub2, sub3. Write a class to define the marklist. Write a constructor to initialize all the variables. Write a method called Result(int min), where, min indicates the minimum marks for “Pass”, if the candidate secures marks in each subject. This method will declare a candidate as “Pass”, if the candidate secures marks in each subject greater than or equal to min. Write another overloading method Result(int min, int merit), where merit indicates that a candidate who gets a “Pass” and average of the three subject equal to or greater than the merit is eligible for getting a scholarship.
5. Declare a class called coordinate to represent 3 dimensional Cartesian coordinates (x, y and z). Define following methods:
   - constructor
   - display, to print values of members
   - add coordinates, to add three such coordinate objects to produce a resultant coordinate object.
   - main, to show use of above methods.
6. It is required to compute SPI (semester performance index) of n students of your college for their registered subjects in a semester. Declare a class called student having following data members: id_no, no_of_subjects_registered, subject_code, subject_credits, grade_obtained and spi.
   - Define constructor and calculate_spi methods.
• Define main to instantiate an array for objects of class student to process data of n students to be given as command line arguments.

7. Design a class named Fan to represent a fan. The class contains:
• Three constants named SLOW, MEDIUM and FAST with values 1,2 and 3 to denote the fan speed.
• An int data field named speed that specifies the speed of the fan (default SLOW).
• A boolean data field named f_on that specifies whether the fan is on(default false).
• A double data field named radius that specifies the radius of the fan (default 4).
• A data field named color that specifies the color of the fan (default blue).
• A no-arg constructor that creates a default fan.
• A parameterized constructor initializes the fan objects to given values.
• A method named display() will display description for the fan. If the fan is on, the display() method displays speed, color and radius. If the fan is not on, the method returns fan color and radius along with the message “fan is off”.

Write a test program that creates two Fan objects. One with default values and the other with medium speed, radius 6, color brown, and turned on status true. Display the descriptions for two created Fan objects.

8. Define the Rectangle class that contains:
• Two double fields x and y that specify the center of the rectangle
• the data field width and height,
• A no-arg constructor that creates the default rectangle with (0,0) for (x,y) and 1 for both width and height.
• A parameterized constructor creates a rectangle with the specified x, y, height and width.
• A method getArea() that returns the area of the rectangle.
• A method getPerimeter() that returns the perimeter of the rectangle.
• A method contains(double x, double y) that returns true if the specified point (x,y) is inside this rectangle.

Write a test program that creates two rectangle objects. One with default values and other with user specified values. Test all the methods of the class for both the objects.

Practical – 3 (Aim: To implement concept of String)
1. Create a StringBuffer object and illustrate following:
   • how to append and insert characters. Display the capacity and length of the string buffer
   • reverse() methods.
   • Displays the substring formed by the last six characters of a string.

2. Print out the alphabet and ascii value on one line of output.
   e.g. A 65 B 66 C 67 ....Z 90

3. Write a program to find length of string and print second half of the string.

4. Write a program to accept a line and check how many consonants and vowels are there in line.

5. Write a program to count the number of words that start with capital letters.
6. Write a program to find that given number or string is palindrome or not.
7. Create a class which asks the user to enter a sentence, and it should display count of each vowel type in the sentence. The program should continue till user enters a word “quit”. Display the total count of each vowel for all sentences.
8. Write an interactive program to print a string entered in a pyramid form. For instance, the string “stream” has to be displayed as follows:
   
   S
   S t
   S t r
   S t r e
   S t r e a
   S t r e a m

9. Write an interactive program to print a diamond shape. For example, if user enters the number 3, the diamond will be as follows:
   
   *
   **
   ***
   **
   *

Practical – 4 (Aim: To implement concept of Inheritance)

1. Define an abstract class called Polygon. Provide a constructor which takes an array of Cartesian Point as parameter. Also provide method called perimeter, which calculates and returns the perimeter of the Polygon. Declare abstract method area for this class. Also define a method called move, which takes two parameters x and y to specify the destination for the first point of the Polygon, and overload to make it work for Cartesian Point as a parameter. Now update the classes Triangle and Rectangle in the exercise 8 above, to be a subclass of the Polygon class. Write appropriate class with main method to test the polymorphism in the area method.
2. Declare a class called book having author_name as private data member. Extend book class to have two sub classes called book_publication & paper_publication. Each of these classes have private member called title. Write a complete program to show usage of dynamic method dispatch (dynamic polymorphism) to display book or paper publications of given author. Use command line arguments for inputting data.
3. The Transport interface declares a deliver () method. The abstract class Animal is the super class of the Tiger, Camel, Deer and Donkey classes. The transport interface is implemented by the Camel and Donkey classes. Write a test program that initialize an array of four Animal objects. If the object implements the Transport interface, the deliver () method is invoked.
4. Describe abstract class called Shape which has three subclasses say Triangle, Rectangle, Circle. Define one method area() in the abstract class and override this area() in these three subclasses to calculate for specific object i.e. area() of Triangle subclass should calculate area of triangle etc. Same for Rectangle and Circle.
5. The abstract Vegetable class has three subclasses named Potato, Brinjal and Tomato. Write an application that demonstrates how to establish this class
hierarchy. Declare one instance variable of type String that indicates the color of a vegetable. Create and display instances of these objects. Override the toString() method of Object to return a string with the name of the vegetable and its color.

6. Declare a class called employee having employee_id and employee_name as members. Extend class employee to have a subclass called salary having designation and monthly_salary as members. Define following:
   - Required constructors
   - A method to find and display all details of employees drawing salary more than Rs. 20000/-. 
   - Method main for creating an array for storing these details given as command line arguments and showing usage of above methods.

7. Write a program to demonstrate combination of both types of inheritance as shown in figure 1. i.e. hybrid inheritance

![Figure 1](image1)

8. Write a program to demonstrate the multipath inheritance for the classes having relations as shown in figure 2. A->(B,C)->D

![Figure 2](image2)

Practical – 5 (Aim: To implement concept of Interface and Package)

1. Write a program that illustrates interface inheritance. Interface P is extended by P1 and P2. Interface P12 inherits from both P1 and P2. Each interface declares one constant and one method. Class Q implements P12. Instantiate Q and invoke each of its methods. Each method displays one of the constants.
2. Write a program that demonstrates the instanceof operator. Declare interfaces I1 and I2. Interface I3 extends both of these interfaces. Also declare interface I4. class X implements I3. Class W extends X and implements I4. Create an object of class W. Use the instanceof operator to test if that object implements each of the interfaces and is of type X.
3. Write an application using user-built package that also imports another user-built package.
4. Write a program that defines CircleArea class with three constructors. The first form accepts no arguments. The second accepts one double value for radius. The third form accepts any two arguments.
5. Write an application that illustrates method overriding. Class Aclass is extended by Bclass. Each of these classes defines a display() method that outputs the string “I am in Aclass” or “I am in Bclass”, respectively. Declare an array to hold six
Aclass objects. Initialize the elements of the array with a mix of Aclass and Bclass objects. Execute a program loop to invoke the display() method of each object.

6. Describe abstract class called Shape which has three subclasses say Triangle, Rectangle and Circle. Define one method area() in the abstract class and override this area() in these three subclasses to calculate for specific object i.e. area() of Triangle subclass should calculate area of triangle etc. Same for Rectangle and Circle.

7. Write a program to define abstract class, with two methods addition() and subtraction(). addition() is abstract method. Implement the abstract method and call that method using a program(s).

Practical – 6 (Aim: To implement concept of Exception)

1. Make a program using try-catch blocks. If user enters only one command line argument than throw an exception and if it enters two arguments and divides the first argument by first. If the second argument is zero than catch the appropriate exception.

2. Write an application that contains a method named average() has one argument that is an array of strings. It converts these to double values and returns their average. The method generates a NullPointerException, if an array elements is null or a NumberFormatException, if an element is incorrectly formatted. Include throws statement in method declaration.

3. Write an application that generates custom exception if first argument from command line argument is 0.

4. Write a program that accepts the fully qualified name of a class as its argument. Compute and display how many super classes exist for that class. If a ClassNotFoundException occurs, catch it and provide an error message for the user.

5. A marklist containing reg.no and marks for a subject is given. If the marks are <0, user-defined IllegalMarkException is thrown out and handled with the message “Illegal Mark”. For all valid marks, the candidate will be declared as “PASS” if the marks are equal to or greater than 40, otherwise it will be declared as “FAIL”. Write a class called IllegalMarkException.

Practical – 7 (Aim: To implement concept of Thread)

1. Write a program that executes two threads. One thread will print the even numbers and another thread will print odd numbers from 1 to 50.

2. Write a program to print “string1 string2 string3” through a synchronized displaystring() method by using String array.

3. Create a Date Class, and output the date in multiple format such as MM/DD/YYYY September 12, 2006 DD MM YYYY, Using overloaded constructor.

4. Write a program in java that creates a child thread by implementing runnable interface and then apply sleep() method to both main thread and child thread.

5. Write an application that creates and starts three threads. Each thread is instantiated from the same class. It executes a loop with 10 iterations. Each iteration displays string “HELLO”, sleeps for 300 milliseconds. The application waits for all the threads to complete & displays the message “Good Bye...”

6. Write an application that executes two threads. One thread displays “Good Morning” every 1000 milliseconds & another thread displays “Good
Afternoon" every 3000 milliseconds. Create the threads by implementing the Runnable interface.

7. Write a complete multi-threaded program to meet following requirements:
   - Two threads of same type are to be instantiated in the method main.
   - Each thread acts as a producer as well as a consumer.
   - A shared buffer can store only one integer information along with the source & destination of the information at a time.
   - The information produced is to be consumed by appropriate consumer.
   - Both producers produce information for both consumers.
   - Each thread produces 5 information.

Practical – 8 (Aim: To implement concept of Input/Output)
1. Write an application that writes the first 15 numbers of the Fibonacci series to a file. A second application that reads this data from a file and displays it.(for this application take filename as a command line argument)
2. Write a program to display the bytes of a file in reverse sequence. Provide the name of the file as a command line argument(Use RandomAccessFile)
3. Write a program to create directories (/home/abc/bcd/def/ghi/jkl) in the home directory /home/abc and list the files and directories showing file/directory, file size. Read-write-execute permissions. Write destructor to destroy the data of a class.
4. Write a standalone program that takes a command line parameter and after ensuring the file does not exist, create a file with that name Read customer name(string) and phone no(integer) form keyboard and store it in file.
5. Write a java program that finds the largest file in the directory given as command line argument
6. Write a program that counts the no. of words in a text file. The file name is passed as a command line argument. The program should check whether the file exists or not. The words in the file are separated by white space characters.
7. Create a class called Student. Write a student manager program to manipulate the student information from files by using FileInputStream and FileOutputStream
8. Refine the student manager program to manipulate the student information from files by using the BufferedReader and BufferedWriter
9. Refine the student manager program to manipulate the student information from files by using the DataInputStream and DataOutputStream. Assume suitable data

Practical – 9 (Aim: To implement concept of UML Diagram)
1. Create following UML Diagrams for Library Management System
   - Class Diagram
   - State Diagram
   - Activity Diagram
   - Sequence Diagram
   - Use case Diagram
2. Prepare a class diagram for given group of classes using multiplicity, generalization, association concepts. And add at least 5-7 attributes and 3-
5 operations for particular class Page, Shape, Point, Line, Arc, Ellipse, Rectangle, Circle.

3. Prepare a class diagram for given group of classes using multiplicity, generalization, association concepts. And add at least 5-7 attributes and 3-5 operations for particular class. City, Airport, Airline, Pilot, Flight, Plane, Seat, Passenger.

4. Categorize the following relationships into generalization, aggregation or association.
   [A] A country has a capital city
   [B] A dining philosopher uses a fork
   [C] A file is an ordinary file or a directory file
   [D] Files contains records
   [E] A polygon is composed of an ordered set of points
   [F] A drawing object is text, a geometrical object, or a group
   [G] A person uses a computer language on a object
   [H] Modems and keyboards are input/output devices
   [I] Classes may have several attributes
   [J] A person plays for a team in a certain year
   [K] A route connects two cities
   [L] A student takes a course from a professor

5. Prepare a state diagram for an interactive diagram editor for selecting and dragging objects.

6. Prepare a use case diagram and sequence diagram for a computer email system.

7. Prepare an activity diagram for computing a restaurant bill, there should be charge for each delivered item. The total amount should be subject to tax and service charge of 18% for group of six and more. For smaller groups there should be a blank entry. Any coupons or gift certificates submitted by the customer should be subtracted.

8. Prepare a sequence diagram for issuing a book in the library management system.

ALL THE BEST
List of Assignments

Assignment – 1 Basics of JAVA
1. Explain features of JAVA.
2. List OOP characteristics and describe inheritance with examples.
3. JVM is platform dependent. Justify.
4. Applet and Application Differentiate it.
5. Explain JDK, JRE, JVM and JIT.

Assignment – 2 Strings and Array
1. String class v/s StringBuffer class.
2. Explain Arrays with Example.
3. Explain various String Functions.
4. Explain Various StringBuffer Functions.

Assignment – 3 Class, Objects and Method
1. What is constructor? Explain constructor overloading with example.
2. Explain Following:
   a) Garbage Collection
   b) Short Circuit Operators.
   c) Shift Operators
3. Explain Dynamic method dispatch with proper example.
4. There is no destructor in Java. Justify.
5. Define and write a program to differentiate between pass by value and pass by reference.

Assignment – 4 Inheritance and Interface
1. Explain following with example:
   a) Super keyword.
   b) Final keyword
   c) Static keyword
   d) this keyword
   e) instanceof and volatile
   f) transient keyword
2. Explain Overloading and Overriding with example.
3. Explain interface and its usage.
4. Define polymorphism with its need. Define and explain static and dynamic binding using program.
5. How do interfaces support polymorphism?
6. What is an inner class? Explain it with example.
7. Explain Cosmic superclass and its methods.
8. Explain Ragged Array.

Assignment – 5 Package and Exception Handling
1. What is package? Explain steps to create package with example.
2. List out all packages with short description.
3. Enlist and explain the difference between error and exception.
4. What is an Exception? Explain Exception handling in JAVA. Explain types of Exceptions.
5. Give some of Built-in exceptions in java. Also give difference between throw and throws. (Checked and Unchecked Exceptions)
6. Explain & illustrate by examples use of final, finally and method finalize.

Assignment – 6 Multithreaded Programming
1. What is Multithreading? Explain Life Cycle of Thread with example.
2. Explain thread Synchronization with join() and isAlive() methods.
3. Explain wait, notify, synchronized and native methods.
4. Explain Thread States and Transitions.

Assignment – 7 IO Programming and Networking
1. Differentiate the following: Text I/O v/s Binary I/O.
2. Compare byte streams and character streams.
3. Explain File constructors, any two methods of class File and method seek.
4. Explain Buffers and Character Array Class.
5. Explain Network programming in Java.
6. Explain URI Class and Datagrams.

Assignment – 8 Collection Class
2. Illustrate by example – generic programming.
3. Define generics in java. Write a program to demonstrate generic interface and generic method.
4. Answer following:
   a) Collection in JAVA.
   b) Features of Map Interface.
5. Compare List, Set and Map interfaces. Also compare ArrayList, TreeSet and HashMap classes in java.
6. Explain utility class Hashtable with example.
7. Differentiate between Vector and ArrayList
8. Differentiate between Enumeration and Iterator.

Assignment – 9 Object Oriented Modeling
1. Explain object oriented and modeling.
2. Explain Association and Links.
3. Explain Aggregation and Composition.
4. Explain Generalization and Inheritance.
5. Explain State, Event, Transition and Condition.
6. Explain Class diagram and State diagram.
7. Explain Use case, Sequence and Activity diagram.