21CS408PC	JAVA PROGRAMMING LAB	L	T	P	C
		0	0	2	1

COURSE OBJECTIVES: Students will be able

- To write programs using abstract classes.
- To write programs for solving real world problems.
- To write exception handling in java.
- To introduce java compiler and eclipse platform.
- To impart hands-on experience with java programming.

COURSE OUTCOMES: After Completion of the course, students will be able to

CO1: develop programs for solving real world problems using java collection framework.
CO2: write programs using abstract classes.
CO3: write multithreaded programs.
CO4: create GUI programs using swing controls in Java.

Note:

- Use LINUX and MySQL for the Lab Experiments. Though not mandatory, encourage the use of the Eclipse platform.
- The list suggests the minimum program set. Hence, the concerned staff is requested to add more problems to the list as needed.

LIST OF EXPERIMENTS:

1. Use Eclipse or Net bean platform and acquaint with the various menus. Create a test project, add a test class, and run it. See how you can use auto suggestions, auto fill. Try code formatter and code refactoring like renaming variables, methods, and classes. Try debug step by step with a small program of about 10 to 15 lines which contains at least one if else condition and a for loop.
2. Write a Java program that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +, -,*, % operations. Add a text field to display the result. Handle any possible exceptions like divided by zero.
3. i) Develop an applet in Java that displays a simple message.
   ii) Develop an applet in Java that receives an integer in one text field, and computes its factorial Value and returns it in another text field, when the button named “Compute” is clicked.
4. Write a program to demonstrate user defined exception handling procedures.
5. Write a Java program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num 2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a Number Format Exception. If Num2 were Zero, the program would throw an Arithmetic Exception. Display the exception in a message dialog box.

6. Write a Java program that implements a multi-thread application that has three threads. First thread generates random integer every 1 second and if the value is even, second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of cube of the number.

7. Write a Java program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green with radio buttons. On selecting a button, an appropriate message with "Stop" or "Ready" or "Go" should appear above the buttons in selected color. Initially, there is no message shown.

8. Write a Java program to create an abstract class named Shape that contains two integers and an empty method named print Area (). Provide three classes named Rectangle, Triangle, and Circle such that each one of the classes extends the class Shape. Each one of the classes contains only the method print Area () that prints the area of the given shape.

9. Suppose that a table named Table.txt is stored in a text file. The first line in the file is the header, and the remaining lines correspond to rows in the table. The elements are separated by commas. Write a java program to display the table using Labels in Grid Layout.

10. Write a Java program that handles all mouse events and shows the event name at the center of the window when a mouse event is fired (Use Adapter classes).

11. Write a Java program that loads names and phone numbers from a text file where the data is organized as one line per record and each field in a record are separated by a tab (\t). It takes a name or phone number as input and prints the corresponding other value from the hash table (hint: use hash tables).

12. Write a Java program that correctly implements the producer – consumer problem using the concept of inter-thread communication.

13. Write a Java program to list all the files in a directory including the files present in all its subdirectories.

REFERENCE BOOKS

1. P. J. Deitel and H. M. Deitel, Java for Programmers, 10th Edition Pearson education.
2. Bruce Eckel, Thinking in Java, Pearson Education, 2006
3. D. S. Malik and P. S. Nair, Java Programming,Cengage Learning.
4. Horstmann and G Cornell, Core Java, Volume 1, 9th edition, Cay S. Pearson, 2013