TOPSTECHNOLOGIES
Training | Outsourcing | Placement | Study Abroad

Java Assignments

Module – 2 (Core Java)

- Write a Java program to Take three numbers from the user and print the greatest number.
- Write a Java program that takes the user to provide a single character from the alphabet. Print Vowel or Consonant, depending on the user input. If the user input is not a letter (between a and z or A and Z), or is a string of length > 1, print an error message.
- Write a Java program that takes a year from user and print whether that year is a leap year or not. B19. Write a program in Java to display the first 10 natural numbers using while loop.
- Write a program in Java to input 5 numbers from keyboard and find their sum and average using for loop.
- Write a program in Java to display the pattern like right angle triangle with a number.

1

12

123

1234

12345

- Write a program in Java to make such a pattern like right angle triangle with number increased by 1 The pattern like:

1

2 3

4 5 6

7 8 9 10

- Write a Java program that reads a positive integer and count the number of digits the number.

Input an integer number less than ten billion: 125463

Number of digits in the number: 6

- Write a Java program to count the letters, spaces, numbers and other characters of an input string.
- Write a Java program to print the ASCII value of a given character.
- Write a Java program that accepts an integer (n) and computes the value of n+nn+nnn. Input number: 5

5 + 55 + 555

- Write a Java program to display the system time.
- Write a Java program to print numbers between 1 to 100 which are divisible by 3, 5 and by both.
- W.A.J.P to get the character at the given index within the String. Original String = Tops Technologies! The character at position 0 is T, The character at position 10 is o
- W.A.J.P to concatenate a given string to the end of another string.
- W.A.J.P to compare a given string to the specified character sequence. Comparing topsint.com and topsint.com: true Comparing Topsint.com and topsint.com: false
- W.A.J.P to check whether a given string ends with the contents of another string. "Java Exercises" ends with "se"? False "Java Exercise" ends with "se"? True
- W.A.J.P to check whether a given string starts with the contents of another string. Red is favorite color. Starts with Red? True Orange is also my favorite color. Starts with Red? False I3.
- W.A.J.P to find all interleaving of given strings.

The given strings are: WX YZ

The interleaving strings are: YWZX WYZX YWXZ WXYZ YZWX WYXZ

- W.A.J.P to find the second most frequent character in a given string. The given string is: successes The second most frequent char in the string is: c
- Create a class named 'Print Number' to print various numbers of different data types by creating different methods with the same name 'printn' having a parameter for each data type.
- Create a class to print an integer and a character with two methods having the same name but different sequence of the integer and the character parameters. For

example, if the parameters of the first method are of the form (int n, char c), then that of the second method will be of the form (char c, int n).

- Create a class to print the area of a square and a rectangle. The class has two methods with the same name but different number of parameters. The method for printing area of a rectangle has two parameters which are length and breadth respectively while the other method for printing area of square has one parameter which is side of square.
- Create a class with a method that prints "This is a parent class" and its subclass with another method that prints "This is child class". Now, create an object for each of the class and call 1 - method of parent class by object of parent class 2 - method of child class by object of child class 3 - method of parent class by object of child class
- Create a class named 'Member' having the following members:
  1. Data members
  2. Name
  3. Age
  4. Phone number
  5. Address
  6. Salary

It also has a method named 'printSalary' which prints the salary of the members.

- Create a class named 'Rectangle' with two data members 'length' and 'breadth' and two methods to print the area and perimeter of the rectangle respectively. Its constructor having parameters for length and breadth is used to initialize the length and breadth of the rectangle. Let class 'Square' inherit the 'Rectangle' class with its constructor having a parameter for its side (suppose s) calling the constructor of its parent class as 'super (s, s)'. Print the area and perimeter of a rectangle and a square.
- Write a program to print the area and perimeter of a triangle having sides of 3, 4 and 5 units by creating a class named 'Triangle' without any parameter in its constructor.
- Print the sum, difference and product of two complex numbers by creating a class named 'Complex' with separate methods for each operation whose real and imaginary parts are entered by user.
- Create an abstract class 'Parent' with a method 'message'. It has two subclasses each having a method with the same name 'message' that prints "This is first subclass" and "This is second subclass" respectively. Call the methods 'message' by creating an object for each subclass.
- Create an abstract class 'Bank' with an abstract method 'getBalance'. $100, $150 and $200 are deposited in banks A, B and C respectively. 'BankA', 'BankB' and 'BankC'

are subclasses of class 'Bank', each having a method named 'getBalance'. Call this method by creating an object of each of the three classes.

- We have to calculate the percentage of marks obtained in three subjects (each out of 100) by student A and in four subjects (each out of 100) by student B. Create an abstract class 'Marks' with an abstract method 'getPercentage'. It is inherited by two other classes 'A' and 'B' each having a method with the same name which returns the percentage of the students. The constructor of student A takes the marks in three subjects as its parameters and the marks in four subjects as its parameters for student B. Create an object for each of the two classes and print the percentage of marks for both the students.
- Write a program to print the factorial of a number by defining a method named 'Factorial'. Factorial of any number n is represented by n! And is equal to 1*2*3*. *(n-1) *n. E.g.-

4! = 1*2*3*4 = 24

3! = 3*2*1 = 6

2! = 2*1 = 2

Also, 1! = 1

0! = 0

- We have to calculate the area of a rectangle, a square and a circle. Create an abstract class 'Shape' with three abstract methods namely 'RectangleArea' taking two parameters, 'SquareArea' and 'CircleArea' taking one parameter each. The parameters of 'RectangleArea' are its length and breadth, that of 'SquareArea' is its side and that of 'CircleArea' is its radius. Now create another class 'Area' containing all the three methods 'RectangleArea', 'SquareArea' and 'CircleArea' for printing the area of rectangle, square and circle respectively. Create an object of class 'Area' and call all the three methods. I3. Write a program which will ask the user to enter his/her marks (out of 100). Define a method that will display grades according to the marks entered as below:

Marks Grade

1-100	A
31-90	B
1-80	B
1-70	C
1-60	D
41-50	DD

40 Fail

- Create a class named 'Shape' with a method to print "This is this is shape". Then create two other classes named 'Rectangle', 'Circle' inheriting the Shape class, both having a method to print "This is rectangular shape" and "This is circular shape" respectively. Create a subclass 'Square' of 'Rectangle' having a method to print "Square is a rectangle". Now call the method of 'Shape' and 'Rectangle' class by the object of 'Square' class.
- W.A.J. P to demonstrate try catch block,
- Take two numbers from the user and perform the division operation and handle Arithmetic Exception. O/P- Enter two numbers: 10 0

Exception in thread main java.lang.ArithmeticException:/ by zero

- W.A.J. P to demonstrate multiple catch blocks, (one is to handle divide by zero exception and another one is to handle

ArrayIndexOutOfBoundException) int a [] =new int [5]; a [5]=30/0;

- W.A.J. P to implement the above program (pro.no-B27) using nesting of try-catch block. try {

try

{//code}

catch (Exception e)

{//code}

catch (Exception e)

{//code}

- W.A.J. P to demonstrate try catch block, take two numbers from the user by Command line argument and perform the division operation and handle Arithmetic

O/P-

Exception in thread main java. Lang. Arithmetic Exception:/ by zero

- W.A.J.P to create the validate method that takes integer value as a parameter. If the age is less than 18, then throw an Arithmetic Exception otherwise print a message welcome to vote.

O/P- Enter your age: 16

Exception in thread main java. Lang. Arithmetic Exception: not valid

- W.A.J.P to create a custom exception if Customer withdraw amount which is greater than account balance then program will show custom exception otherwise amount

will deduct from account balance. Account balance is: 2000 Enter withdraw amount: 2500

Sorry, insufficient balance, you need more 500 Rs. To perform this transaction.

- W.A.J.P to create a class Student with attributes roll no, name, age and course. Initialize values through parameterized constructor. If age of student is not in between 15 and 21 then generate user defined exception "AgeNotWithinRangeException". If name contains numbers or special symbols raise exception "NameNotValidException". Define the two exception classes.
- W.A.J. P to create one thread by implementing Runnable interface in

Class.

- W.A.J. P to create one thread by extending Thread class in another Class.
- W.A.J.P to create 2 threads and execute that threads by providing sleep time as 2000ms and check the execution.
- W.A.J.P to start the same Thread twice by calling start () method twice. Test ThreadTwice1 t1=new TestThreadTwice1(); t1.start (); t1.start ();
- W.A.J.P to create 2 threads and make one thread as Daemon Thread by using set Daemon () method of Thread class and check whether the thread is set daemon or not by using is Daemon () method.

  TestDaemonThread2 t1=new TestDaemonThread2();

  TestDaemonThread2 t2=new TestDaemonThread2(); t1.start();

  t1.setDaemon(true);//will throw exception here t2.start();

Write a Java program to create a new array list, add some colors (string) and print out the collection.

- Write a Java program to iterate through all elements in an array list.
- Write a Java program to insert an element into the array list at the first position.
- Write a Java program to retrieve an element (at a specified index) from a given array list.
- Write a Java program to update specific array element by given element.
- Write a Java program to remove the third element from an array list.
- Write a Java program to search an element in an array list.
- Write a Java program to sort a given array list.
- Write a Java program to copy one array list into another.
- Write a Java program to shuffle elements in an array list.
- Write a Java program to append the specified element to the end of a hash set.

- Write a Java program to iterate through all elements in a hash list.
- Write a Java program to get the number of elements in a hash set.
- Write a Java program to associate the specified value with the specified key in a Hash Map.
- Write a Java program to count the number of key-value (size) mappings in a map.
- Write a Java program to reverse elements in an array list.
- Write a Java program to extract a portion of an array list.
- Write a Java program to compare two array lists.
- Write a Java program of swap two elements in an array list.
- Write a Java program to join two array lists.
- Write a Java program to convert a hash set to an array.
- Write a Java program to convert a hash set to a List/Array List.
- Write a Java program to check whether a map contains key-value mappings (empty) or not.
- Write a Java program to increase the size of an array list.
- Write a Java program to replace the second element of an Array List with the specified element.
- Write a Java program to print all the elements of an Array List using the position of the elements.
- Write a Java program to compare two sets and retain elements which are same on both sets.
- Write a Java program to get a collection view of the values contained in this map.