

Type 2: Java Programs Based on the Core Concepts of Java

1. WAP to illustrate Constructor Overloading and Constructor Chaining.
2. Write a Java Singleton class example using private constructor.
3. WAP that shows how Exception terminates java program.
4. WAP that creates java custom Exception.
5. WAP of Exception Handling which is not having catch block.
6. WAP for java daemon threads.
7. WAP that illustrates java thread joins.
8. Write a simple generics class example with two type parameters.
9. How to iterate through collection objects? Write a Program.
10. How implement bounded types (implements an interface) with generics? Write a Program.
11. How to remove an element from collection using Iterator object? Write a Program.
12. How to read all elements in ArrayList by using iterator? Write a Program.
13. How to shuffle elements in ArrayList? Write a Program.
14. How to initialise or call a static member class? Write a Program.
15. How to implement a static member interface? Write a Program.
16. How to access top level class variables from static member classes? Write a Program.
17. How to initialise non-static member (local class) class? Write a Program.
18. How to create Java URL object. Write a Program.
19. How to get my machine or local host's Host name in Java? Write a Program.
20. How to get list of all IPs of a given Host in Java? Write a Program.

21. How to get list of all file names from a folder in java? Write a Program.
22. How to get list of all file objects from a folder in java? Write a Program.
23. How to filter the files by file extensions and show the file names? Write a Program.
24. How to write or store data into temporary file in java? Write a Program.
25. How to write string content to a file in java? Write a Program.
26. How to write byte content to a file in java? Write a Program.
27. How to get file last modified time? Write a Program.
28. How to read input from java console in java? Write a Program.
29. Write an example for JDBC prepared statement with ResultSet. Write a Program.
30. How to get primary key value (auto-generated keys) from inserted queries using JDBC?
31. How to insert an image into database table? Write a Program.
32. Write an example for inserting BLOB into table. Write a Program.
33. WAP that illustrates the difference between method overloading and method overriding.
34. WAP that shows the usage of *super* in all the ways of java programming.
35. A super class **Record** has been defined to store the names and ranks of 50 students. Define a sub class **Rank** to find the highest rank along with the name. The details of both classes are given below:

Class name : Record

Data Members / instance variables:

name[] : to store the names of students

rnk[] : to store the ranks of students

Member functions:

Record() : constructor to initialize data members

void readvalues() : to store names and ranks

void display() : displays the names and the corresponding ranks

Class name : Rank

Data Members / instance variables:

index : integer to store the index of the topmost rank

Member functions

Rank() : constructor to invoke the base class constructor and to initialize index to 0.

void highest() : finds the index location of the topmost rank and stores it in index without sorting the array 6

void display() : displays the name and ranks along with the name having the topmost rank.

Specify the class **Record** giving details of the **constructor()**, **void readvalues()**, **void display()**. Using the concept of inheritance, specify the class **Rank** giving details of **constructor()**, **void highest()** and **void display()**.

Write the main method and call other methods appropriately.

36. A super class **Detail** has been defined to store the details of a customer. Define a subclass **Bill** to compute the monthly telephone charge of the customer as per the chart given below:

Number Of Calls	Rate
1 – 100	Only Rental charge
101 – 200	60 paisa per call + rental charge
201 – 300	80 paisa per call + rental charge
Above 300	1 rupee per call + rental charge

The details of both the classes are given below:

Class Name : Detail

Data members / Instance variables:

name : to store the name of the customer.
address : to store the address of the customer.
telno : to store the phone number of the customer.
rent : to store the monthly rental charge

Member functions:

Detail(...) : parameterized constructor to assign values to data members.
void show() : to display the detail of the customer.

Class Name : Bill

Data members / Instance variables:

n : to store the number of calls.
amt : to store the amount to be paid by the customer.

Member functions:

Bill(...) : parameterized constructor to assign values to data members of both classes and to initialize amt = 0.0.

void cal() : calculates the monthly telephone charge as per the charge given above.

void show() : to display the detail of the customer and amount to be paid.

Specify the class **Detail** giving details of the **constructor()** and **void show()**. Using the concept of inheritance, specify the class **Bill** giving details of the **constructor()**, **void cal()** and **void show()**.

The main function and algorithm need not be written.

37. WAP that illustrates anonymous inner class.
38. WAP that illustrates instance variable hiding.
39. Write a Java program to merge two files alternatively into third file.
40. WAP that illustrate this, static and final keywords.
41. WAP for Method Overloading with Autoboxing and Widening in Java.
42. Differentiate throw and throws using java program.
43. Write a Java program to find IP address of your computer.
44. WAP that shows Java Object Creation of Inherited Class.
45. WAP of checked and unchecked exception in java.

46. WAP that implements copy constructor in java.
47. WAP for implementing assert in java.
48. WAP that override toString() method in java.
49. WAP for Remote Method Invocation.
50. Write a Java method to display the first 50 pentagonal numbers. Note: A pentagonal number is a figurate number that extends the concept of triangular and square numbers to the pentagon, but, unlike the first two, the patterns involved in the construction of pentagonal numbers are not rotationally symmetrical.