1. WAP to display a message. "Let us start java programming".
2. WAP to find the sum of the two numbers.
3. WAP to understand the effect of print() and println() methods and the backslash codes.
4. WAP to observe the effects of various bitwise operators.
5. WAP implement the arithmetic operators.
6. WAP implement the relational operators.
7. WAP implement the assignment operators.
8. WAP implement the increment operators.
9. WAP implement the decrement operators.
10. WAP implement the logical operators.
11. WAP implement the conditional operator.
12. WAP implement the following math functions
13. sqrt(x).
14. pow(x).
15. sin(x)
16. cos(x)
17. tan(x)
18. ceil(x)
19. floor(x)
20. rint(x)
21. abs(a)
22. max(a,b)
23. min(a,b)
24. WAP to implement only if statement.
25. WAP to implement if­­­\_\_ else statement.
26. WAP to implement nested if­­­\_\_else statement.
27. WAP to implement else if ladder.
28. WAP to test the the given number is positive or negative.
29. WAP to implement if­­­\_\_ else statement.
30. WAP to display numbers from 1 to 100.Using do\_ while loop.
31. WAP to display numbers from 1 to 100.Using while loop.
32. WAP to display numbers from 1 to 100.Using for loop.
33. WAP to display stars in a triangular form –a single star in1st line, two stars in 2nd line, three stars in 3rd line and so on.
34. WAP to implement switch statement.
35. WAP to display numbers from 100 to 1in descending order .Using for loop.
36. WAP to display numbers from 1 to 10.Using infinite for loop.
37. WAP to accept and display a character from the keyboard. Using Buffered Reader class.
38. WAP to accept and display a string or a name from the keyboard. Using Buffered Reader class
39. WAP to accept and display an integer from the keyboard. Using Buffered Reader class
40. WAP to accept and display a float number from the keyboard. Using Buffered Reader class
41. WAP to accept and display employee details. Using Buffered Reader class
42. WAP accept a year from the keyboard and test if it is leap or not.
43. WAP accept any two number from the keyboard and display their sum .
44. WAP to accept and display a character from the keyboard. Using scanner class.
45. WAP to accept and display a string or a name from the keyboard. Using scanner class.
46. WAP to accept and display an integer from the keyboard. Using scanner class.
47. WAP to accept and display a float number from the keyboard. Using scanner class.
48. WAP to accept and display employee details. Using scanner class.
49. WAP accept any integer from the keyboard and test if it is even or not.
50. WAP accept any two float number from the keyboard and display their sum.
51. WAP to create a 1D array and read its elements by using a loop display them one by one.
52. WAP to accepts the marks of a students into 1D array from the keyboard and finds total marks and percentage.
53. WAP to take a 2D array and display its elements in the form of a array.
54. WAP to take a 2D array and display its elements in the form of a array also display its transpose.
55. WAP to initialize the person class instance variables in Demo class.
56. WAP to initialize the instance variables directly within the Person1 class.
57. WAP to initialize the instance variables directly within the Person1 class using a default constructor.
58. WAP to initialize the instance variables using a parameterized constructor.
59. WAP a Java program to accept a person’s name and age and display if he is young, middle aged or old to understand the use of methods in a class.
60. WAP for a method without parameters but with return type.
61. WAP for a method without parameters and without return type.
62. WAP for a method with two parameters and return type.
63. WAP to interchange two integers by passing them to swap ( ) method.
64. WAP to add two matrices and display sum matrix.
65. WAP to find factorial value without using recursion.
66. WAP to find factorial value by using recursion.
67. WAP to calculate and display area of circle.
68. WAP to implement single inheritance.
69. WAP to implement multiple inheritance.
70. WAP to implement method overriding.
71. WAP to implement method overloading.
72. WAP to illustrate how to achieve multiple inheritance using multiple interface.