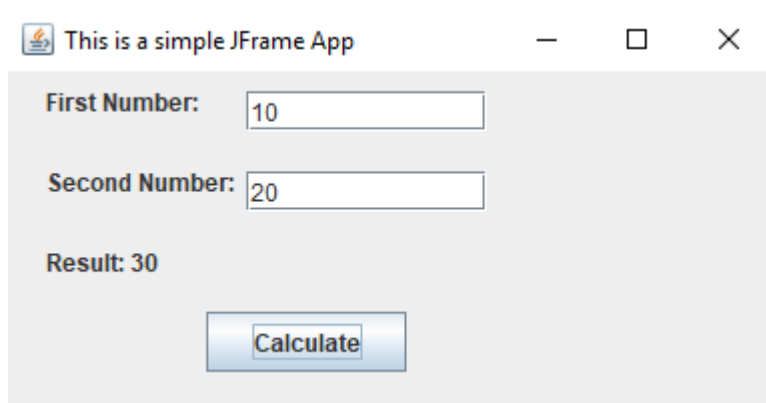# Lab Sheet of Java

1. WAP to find sum of two numbers.
2. WAP to find product of two numbers.
3. WAP to add, subtract, multiply and divide two numbers.
4. WAP to find simple interest. [si=(p*t*r)/100]
5. WAP to area of rectangle. [area=l*b]
6. WAP to find area of circle. [area=pi*r*r] (use pi as constant)
7. WAP to find largest among two numbers.
8. WAP to find smallest among two numbers.
9. WAP to find largest among three numbers.
10. WAP to find smallest among three numbers.
11. WAP to check whether a number is odd or even.
12. WAP to check whether a number is divisible by 7 or not.
13. WAP to check whether a number is exactly by 5 and 10.
14. WAP to check whether a number a number is divisible by 7 but not by 13.
15. WAP to input CP and SP and check profit or loss. Also find profit or loss amount.
16. WAP to typecast the following:
    a) Integer to String
    b) String to Integer
    c) Integer to Double
    d) Double to Integer
    e) String to Double
    f) Double to String
17. WAP to find print numbers from 1 to 10.
18. WAP to find sum of numbers from 5 to 100.
19. WAP to print following series.
    a. 5, 10, 15, 20, …… 50
    b. 1, 4, 9, 16, ….. upto 20 terms.
    c. 100, 98, 96, 94, …………. Upto 10 terms.
20. WAP to print first 15 even numbers.
21. WAP to find sum of odd numbers from 1 to 100.
22. WAP to find factorial of a number.
23. WAP to print following Fibonacci series. 1, 1, 2, 3, 5, 8, …….. upto 15 terms.
24. WAP to print following pattern.

a.	b.	c.	d.
*	*****	1	1
**	****	12	22
***	***	123	333
****	**	1234	4444
*****	*	12345	55555

25. WAP to check whether a number is prime or not.
26. WAP to print prime numbers from 1 to 100.
27. WAP to show the use of ternary operator.
28. Write a program to show the use of switch case statement.
29. Write a program to show the use of auto-increment and auto-decrement operators.
30. Write a program to show the use of break, continue and return.

31. Write a program to define a class with its data members and function members. Use object of this class in main program to access its members.
32. Write a program to define a class named Box which has data **length**, **breadth** and **height** and public functions **ReadData**() for reading data members and **Volume**() to calculate volume of box.
33. Write a program which has class **Book** with data members **book_name, ISBN, author & price** and appropriate function members to read and display data members.
34. Define a class **Rectangle** with data members: **length and breadth**. Initialize its data members with some fixed values (i.e. say 100 and 200 for length and breadth respectively) using a constructor. Write a program to use an object of the class to calculate area of a rectangle.
35. Modify above program to read **length and breadth** of a rectangle in **main()** function and supply them in parameterized constructor to initialize its data members.
36. Write a program of your choice to show the implementation of **this** keyword.
37. Write a program which has two functions with same name, one for addition of two integers and other for addition of three integers.
38. Write a program which has two functions with same name and same number of arguments, one for addition of two integers and other for addition of two double values.
39. Write a program to define a class **Circle** with its data members **pi** and **r** and members functions **getdata()** for initializing data members and **calculate()** for finding area of ac circle. Return result from **calculate()** and display result in **main()** function. Use **pi** as constant.
40. Write a program to implement encapsulation using **getter** and **setter** methods.
41. Create a class named **Person** which has **name & age** as data members and appropriate function members to read and display its data. Create another class **Employee** derived from class **Person** to use features of base class (**single**).
42. Create a class **Polygon** with data members: **dimension1 and dimension2** and a member function: **ReadDimension()** to read data members. Derive two classes **Rectangle** and **Triangle** from **Polygon** class with appropriate member function to calculate area of each rectangle and triangle (**multilevel**).
43. Create a class **Vehicle** with data members: **VNo, no_of_wheel and max_speed**. Derive another class **Passenger** with data member: **no_of_passengers**. Derive two other classes **Bus(with route, fare_per_person and helper_name) and Taxi (with fare_per_km as data member)**. Write a program to use these classes (**multilevel**).
44. Write a program with two classes. Include a function with same name and same signature in each class to illustrate use of function overriding.
45. Create a class **Polygon** with data members to represent two dimensions and **parameterized constructor** to initialize data members. Derive two classes **Rectangle** and **Triangle** from **Polygon** class with appropriate member function to calculate area of each rectangle and triangle.
46. Write a program of your choice to implement **multiple inheritance** using **interface**.
47. Write a program to implement **abstract class and final class** to achieve abstraction.

48. Write a program to demonstrate exception handling using **try, catch and finally** block.
49. Write a program to handle following exceptions:
    a) **Arithmetic**   b) **NullPointer**   c) **ArrayIndexOutOfBound**
    d) **StringIndexOutofBound**   e) **NumberFormatException**
50. Write to program to demonstrate **throw and throws** keyword.
51. Write a program to demonstrate **nested try** block.
52. Write a program to demonstrate **multiple catch** block.
53. Write a program to input any string and convert it to **uppercase and lowercase**.
54. Write a program to demonstrate character extraction using **charAt()** and **getChars()** methods.
55. Write a program to demonstrate string comparison using **equals()** and **compareTo()** methods.
56. Write a program to search any string **indexOf()** and **lastIndexOf()** methods.
57. Write a program to demonstrate modification of string using **substring()**, **replace()**, **concat()** and **trim()** methods.
58. Write a program to demonstrate various methods of **String Buffer** class.
59. Write a program to demonstrate thread by **extending Thread class**.
60. Write a program to demonstrate thread by **implementing Runnable interface**.
61. Write a program to demonstrate **sleep()** and **stop()** methods.
62. Write a program to **get and set priorities** in thread.
63. Write a program to demonstrate **Inter Thread Communication** using Synchronization.
64. Write a program to demonstrate **deadlock condition**.
65. Write a program to **push and pop items** in and from stack.
66. Write a program to demonstrate **vector**.
67. Write a program to demonstrate **hash table**.
68. Write a program to generate **random number** in Java.
69. Write a program to implement **Map, List and Set Interface**.
70. Write a program to demonstrate **Array List, Linked List, Hash Set and Tree Set**.
71. Write a program to demonstrate **Iterator and Comparator** in Collection Framework.
72. Write a **swing program** for the following:
    a)

b)

73. Write a swing program to demonstrate **key and mouse event handling**.
74. Write a program to create **dialog box**.
75. Write a program of your choice to demonstrate basic Java Applet.
76. Write a program to create **database connection** and demonstrate **data manipulation** using **JDBC** (Show Basic **CRUD** operation).
77. Write a program to create a file and demonstrate basic file read and write operation.