Q1

Ans. public class JavaExample

{

int add(int num1, int num2)

{

return num1+num2;

}

int add(int num1, int num2, int num3)

{

return num1+num2+num3;

}

int add(int num1, int num2, int num3, int num4)

{

return num1+num2+num3+num4;

}

public static void main(String[] args)

{

JavaExample obj = new JavaExample();

//This will call the first add method

System.out.println("Sum of two numbers: "+obj.add(10, 20));

//This will call second add method

System.out.println("Sum of three numbers: "+obj.add(10, 20, 30));

//This will call third add method

System.out.println("Sum of four numbers: "+obj.add(1, 2, 3, 4));

}

}

Output:

Sum of two numbers: 30

Sum of three numbers: 60

Sum of four numbers: 10

Q1.

Ans. import java.util.Scanner;

public class JavaProgram

{

public static void main(String args[])

{

int first, second, add, subtract, multiply;

float devide;

Scanner scanner = new Scanner(System.in);

System.out.print("Enter Two Numbers : ");

first = scanner.nextInt();

second = scanner.nextInt();

add = first + second;

subtract = first - second;

multiply = first \* second;

devide = (float) first / second;

System.out.println("Sum = " + add);

System.out.println("Difference = " + subtract);

System.out.println("Multiplication = " + multiply);

System.out.println("Division = " + devide);

}

}

Output of program:

Enter Two Numbers : 12 5

Sum = 17

Difference = 7

Multiplication = 60

Division = 2.4

Q2.

Ans. /\* java Program to find modulus of two numbers \*/

//Save it as ModulusTwoNumbers.java

import java.io.\*;

import java.util.Scanner;

public class ModulusTwoNumbers {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

int firstNumber, secondNumber, mod;

System.out.println("Enter first number : ");

firstNumber = scanner.nextInt();

System.out.println("Enter second number : ");

secondNumber = scanner.nextInt();

mod = firstNumber % secondNumber;

System.out.println("The modulus of "+firstNumber+" and "+secondNumber+" is "+mod);

}

}

**Input:**

Enter first number :

15

Enter second number :

2

**Output:**

The modulus of 15 and 2 is 1

Q3.

#### Ans. Solution of the above problem by this keyword

1. **class** Student{
2. **int** rollno;
3. String name;
4. **float** fee;
5. Student(**int** rollno,String name,**float** fee){
6. **this**.rollno=rollno;
7. **this**.name=name;
8. **this**.fee=fee;
9. }
10. **void** display(){System.out.println(rollno+" "+name+" "+fee);}
11. }
13. **class** TestThis2{
14. **public** **static** **void** main(String args[]){
15. Student s1=**new** Student(111,"ankit",5000f);
16. Student s2=**new** Student(112,"sumit",6000f);
17. s1.display();
18. s2.display();
19. }}

[**Test it Now**](https://www.javatpoint.com/opr/test.jsp?filename=TestThis2)

**Output:**

111 ankit 5000.0

112 sumit 6000.0

Q4.

Ans. class Person {

String name;

String msg;

public Person(String name, String msg) {

this.name = name;

this.msg = msg;

}

void message() {

System.out.println("Inside message method of Person class");

}

}

class Employee extends Person {

String msg;

int empId;

public Employee(String name, int id) {

super(name,"Variable of Person class"); // Superclass constructor call must be first statement

this.empId = id;

this.msg = "Variable of Employee class";

}

void printDetail() {

System.out.println("Name = "+this.name+", empId = "+this.empId);

super.message();

this.message();

System.out.println(super.msg);

System.out.println(this.msg);

}

void message() {

System.out.println("Inside message method of Employee class");

}

public static void main(String [] args) {

Employee emp = new Employee("Rahul", 20);

emp.printDetail();

}

}

**Output:**

Name = Rahul, empId = 20

Inside message method of Person class

Inside message method of Employee class

Variable of Person class

Variable of Employee class

Q4.

Ans. 1. class Animal{ 2. String color="white"; 3. } 4. class Dog extends Animal{ 5. String color="black"; 6. void printColor(){ 7. System.out.println(color);//prints color of Dog class 8. System.out.println(super.color);//prints color of Animal class 9. } 10. } 11. class TestSuper1{ 12.public static void main(String args[]){ 13. Dog d=new Dog(); 14. d.printColor(); 15. }} Test it Now Output: black white

Q5.

Ans. public class CommandLineArguments {

    public static void main(String[] args) {

        int a = Integer.parseInt(args[0]);

        int b = Integer.parseInt(args[1]);

        int sum = a + b;

        System.out.println("Sum is " + sum);

    }

}

Q6.

Ans. public class OverloadedMethod  
{  
public int FunctionName(int x, int y) //Two parameters in the function  
{  
return (x + y); //Returns the sum of the two numbers  
}  
// This function takes three integer parameters  
public int FunctionName(int x, int y, int z)  
{  
return (x + y + z);  
}  
// This function takes two double parameters  
public double FunctionName(double x, double y)  
{  
return (x + y);  
}//Many more such methods can be done with different number of parameters  
// Code used to input the number and  
public static void main(String args[])  
{  
FunctionName s = new FunctionName();  
System.out.println(s.FunctionName(10, 20));  
System.out.println(s. FunctionName(10, 20, 30));  
System.out.println(s. FunctionName(10.5, 20.5));  
}  
}

Q7.

Ans. class Factorial

{

int num, f;

Factorial()

{

f = 1;

}

Factorial(int n)

{

num = n;

}

public int getFactorial()

{

for(int i = 1; i <= num; i++)

{

f = f \* i;

}

System.out.println("Factorial= " + f);

return f;

}

public static void main(int m)

{

Factorial obj = new Factorial();

obj = new Factorial(m);

obj.getFactorial();

}

}

Q8.

Ans. class Car{

public Car()

{

System.out.println("Class Car");

}

public void vehicleType()

{

System.out.println("Vehicle Type: Car");

}

}

class Maruti extends Car{

public Maruti()

{

System.out.println("Class Maruti");

}

public void brand()

{

System.out.println("Brand: Maruti");

}

public void speed()

{

System.out.println("Max: 90Kmph");

}

}

public class Maruti800 extends Maruti{

public Maruti800()

{

System.out.println("Maruti Model: 800");

}

public void speed()

{

System.out.println("Max: 80Kmph");

}

public static void main(String args[])

{

Maruti800 obj=new Maruti800();

obj.vehicleType();

obj.brand();

obj.speed();

}

}

Output:

Class Car

Class Maruti

Maruti Model: 800

Vehicle Type: Car

Brand: Maruti

Max: 80Kmph

Q9.

Ans. //abstract parent class

abstract class Animal{

//abstract method

public abstract void sound();

}

//Dog class extends Animal class

public class Dog extends Animal{

public void sound(){

System.out.println("Woof");

}

public static void main(String args[]){

Animal obj = new Dog();

obj.sound();

}

}

Output:

Woof

Q10.

Ans. interface AnimalEat {

   void eat();

}

interface AnimalTravel {

   void travel();

}

class Animal implements AnimalEat, AnimalTravel {

   public void eat() {

      System.out.println("Animal is eating");

   }

   public void travel() {

      System.out.println("Animal is travelling");

   }

}

public class Demo {

   public static void main(String args[]) {

      Animal a = new Animal();

      a.eat();

      a.travel();

   }

}

## Output

Animal is eating

Animal is travelling

Q11.

Ans.

|  |
| --- |
| // Java program to demonstrate ArithmeticException  class ArithmeticException\_Demo  {      public static void main(String args[])      {          try {              int a = 30, b = 0;              int c = a/b;  // cannot divide by zero              System.out.println ("Result = " + c);          }          catch(ArithmeticException e) {              System.out.println ("Can't divide a number by 0");          }      }  } |

**Output:**

Can't divide a number by 0

Q12.

Ans. There are three ways to compare String in Java:

1. By Using equals() Method
2. By Using == Operator
3. By compareTo() Method

## 1) By Using equals() Method

1. **class** Teststringcomparison1{
2. **public** **static** **void** main(String args[]){
3. String s1="Sachin";
4. String s2="Sachin";
5. String s3=**new** String("Sachin");
6. String s4="Saurav";
7. System.out.println(s1.equals(s2));//true
8. System.out.println(s1.equals(s3));//true
9. System.out.println(s1.equals(s4));//false
10. }
11. }

[**Test it Now**](https://www.javatpoint.com/opr/test.jsp?filename=Teststringcomparison1)

**Output:**

true

true

false

In the above code, two strings are compared using **equals()** method of **String** class. And the result is printed as boolean values, **true** or **false**.

**Teststringcomparison2.java**

1. **class** Teststringcomparison2{
2. **public** **static** **void** main(String args[]){
3. String s1="Sachin";
4. String s2="SACHIN";
6. System.out.println(s1.equals(s2));//false
7. System.out.println(s1.equalsIgnoreCase(s2));//true
8. }
9. }

[**Test it Now**](https://www.javatpoint.com/opr/test.jsp?filename=Teststringcomparison2)

**Output:**

false

true

## 2) By Using == operator

1. **class** Teststringcomparison3{
2. **public** **static** **void** main(String args[]){
3. String s1="Sachin";
4. String s2="Sachin";
5. String s3=**new** String("Sachin");
6. System.out.println(s1==s2);//true (because both refer to same instance)
7. System.out.println(s1==s3);//false(because s3 refers to instance created in nonpool)
8. }
9. }

[**Test it Now**](https://www.javatpoint.com/opr/test.jsp?filename=Teststringcomparison3)

**Output:**

true

false

## 3) String compare by compareTo() method

## 3) By Using compareTo() method

1. **class** Teststringcomparison4{
2. **public** **static** **void** main(String args[]){
3. String s1="Sachin";
4. String s2="Sachin";
5. String s3="Ratan";
6. System.out.println(s1.compareTo(s2));//0
7. System.out.println(s1.compareTo(s3));//1(because s1>s3)
8. System.out.println(s3.compareTo(s1));//-1(because s3 < s1 )
9. }
10. }

[**Test it Now**](https://www.javatpoint.com/opr/test.jsp?filename=Teststringcomparison4)

**Output:**

0

1

-1

Q17.

Ans.

|  |
| --- |
| // Java program for simple calculator    import java.io.\*;  import java.lang.\*;  import java.lang.Math;  import java.util.Scanner;  public class BasicCalculator {        public static void main(String[] args)      {          // stores two numbers          double num1, num2;            // Take input from the user          Scanner sc = new Scanner(System.in);            System.out.println("Enter the numbers");            // take the inputs          num1 = sc.nextDouble();            num2 = sc.nextDouble();            System.out.println("Enter the operator (+,-,\*,/)");            char op = sc.next().charAt(0);            double o = 0;            switch (op) {            // case to add two numbers          case '+':                o = num1 + num2;                break;            // case to subtract two numbers          case '-':                o = num1 - num2;                break;            // case to multiply two numbers          case '\*':                o = num1 \* num2;                break;            // case to divide two numbers          case '/':                o = num1 / num2;                break;            default:                System.out.println("You enter wrong input");                break;          }            System.out.println("The final result:");            System.out.println();            // print the final result          System.out.println(num1 + " " + op + " " + num2                             + " = " + o);      }  } |

**Output:**

*Enter the numbers:*

*2*

*2*

*Enter the operator (+,-,\*,/)*

*+*

*The final result:*

*2.0 + 2.0 = 4.0*

Q18.

### Ans. ****Simple Applet****

import java.awt.\*;

import java.applet.\*;

public class Simple extends Applet

{

public void paint(Graphics g)

 {

  g.drawString("A simple Applet", 20, 20);

 }

}

### Example of an Applet

import java.applet.\*;

import java.awt.\*;

public class MyApplet extends Applet

{

int height, width;

public void init()

 {

  height = getSize().height;

  width = getSize().width;

setName("MyApplet");

 }

public void paint(Graphics g)

 {

  g.drawRoundRect(10, 30, 120, 120, 2, 3);

 }

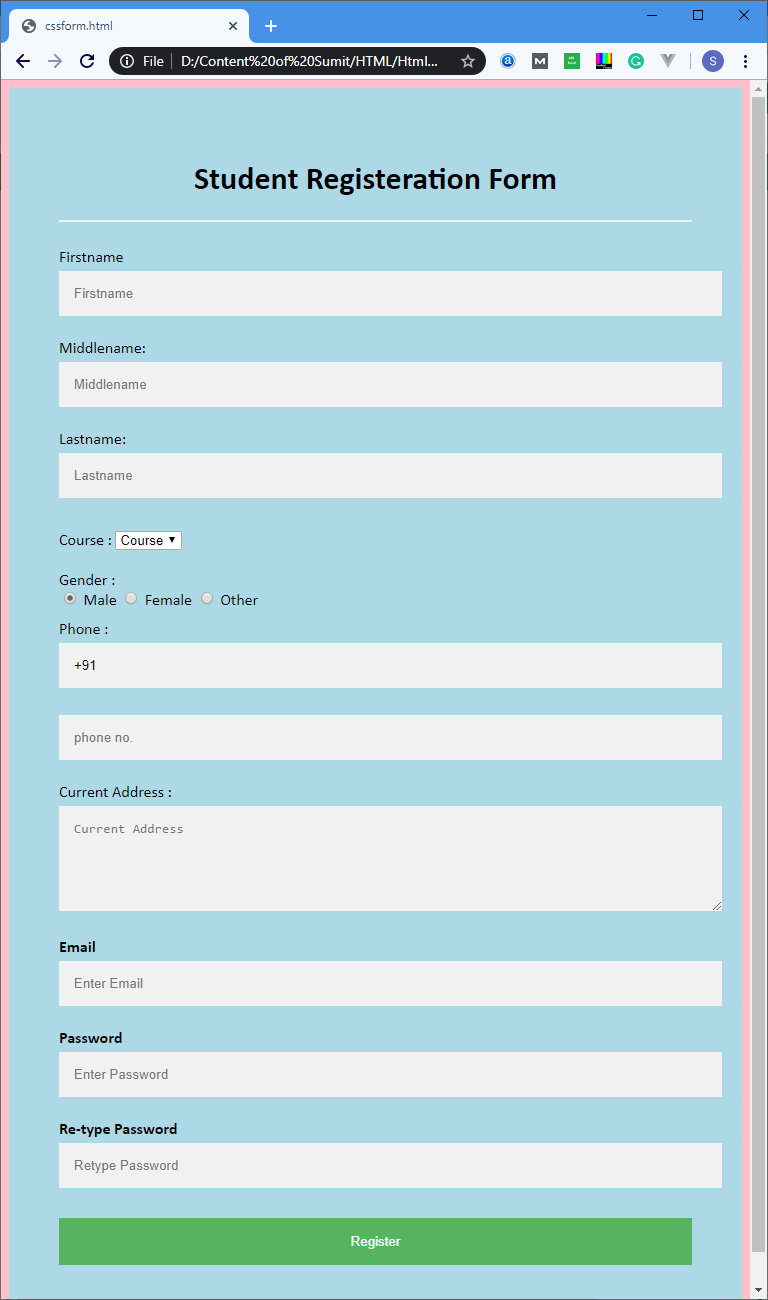
}

Q18.

1. Ans. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<meta** name="viewport" content="width=device-width, initial-scale=1"**>**
5. **<style>**
6. body{
7. font-family: Calibri, Helvetica, sans-serif;
8. background-color: pink;
9. }
10. .container {
11. padding: 50px;
12. background-color: lightblue;
13. }
15. input[type=text], input[type=password], textarea {
16. width: 100%;
17. padding: 15px;
18. margin: 5px 0 22px 0;
19. display: inline-block;
20. border: none;
21. background: #f1f1f1;
22. }
23. input[type=text]:focus, input[type=password]:focus {
24. background-color: orange;
25. outline: none;
26. }
27. div {
28. padding: 10px 0;
29. }
30. hr {
31. border: 1px solid #f1f1f1;
32. margin-bottom: 25px;
33. }
34. .registerbtn {
35. background-color: #4CAF50;
36. color: white;
37. padding: 16px 20px;
38. margin: 8px 0;
39. border: none;
40. cursor: pointer;
41. width: 100%;
42. opacity: 0.9;
43. }
44. .registerbtn:hover {
45. opacity: 1;
46. }
47. **</style>**
48. **</head>**
49. **<body>**
50. **<form>**
51. **<div** class="container"**>**
52. **<center>**  **<h1>** Student Registeration Form**</h1>** **</center>**
53. **<hr>**
54. **<label>** Firstname **</label>**
55. **<input** type="text" name="firstname" placeholder= "Firstname" size="15" required **/>**
56. **<label>** Middlename: **</label>**
57. **<input** type="text" name="middlename" placeholder="Middlename" size="15" required **/>**
58. **<label>** Lastname: **</label>**
59. **<input** type="text" name="lastname" placeholder="Lastname" size="15"required **/>**
60. **<div>**
61. **<label>**
62. Course :
63. **</label>**
65. **<select>**
66. **<option** value="Course"**>**Course**</option>**
67. **<option** value="BCA"**>**BCA**</option>**
68. **<option** value="BBA"**>**BBA**</option>**
69. **<option** value="B.Tech"**>**B.Tech**</option>**
70. **<option** value="MBA"**>**MBA**</option>**
71. **<option** value="MCA"**>**MCA**</option>**
72. **<option** value="M.Tech"**>**M.Tech**</option>**
73. **</select>**
74. **</div>**
75. **<div>**
76. **<label>**
77. Gender :
78. **</label><br>**
79. **<input** type="radio" value="Male" name="gender" checked **>** Male
80. **<input** type="radio" value="Female" name="gender"**>** Female
81. **<input** type="radio" value="Other" name="gender"**>** Other
83. **</div>**
84. **<label>**
85. Phone :
86. **</label>**
87. **<input** type="text" name="country code" placeholder="Country Code"  value="+91" size="2"**/>**
88. **<input** type="text" name="phone" placeholder="phone no." size="10"/ required**>**
89. Current Address :
90. **<textarea** cols="80" rows="5" placeholder="Current Address" value="address" required**>**
91. **</textarea>**
92. **<label** for="email"**><b>**Email**</b></label>**
93. **<input** type="text" placeholder="Enter Email" name="email" required**>**
95. **<label** for="psw"**><b>**Password**</b></label>**
96. **<input** type="password" placeholder="Enter Password" name="psw" required**>**
98. **<label** for="psw-repeat"**><b>**Re-type Password**</b></label>**
99. **<input** type="password" placeholder="Retype Password" name="psw-repeat" required**>**
100. **<button** type="submit" class="registerbtn"**>**Register**</button>**
101. **</form>**
102. **</body>**
103. **</html>**

[**Test it Now**](https://www.javatpoint.com/oprweb/test.jsp?filename=html-registration-form2)

**Output:**



Q20.

Ans.