

Online Java Compiler



Output

```
1 import java.util.*;
2
3 public class Main {
4     public static void main(String[] a){
5         Scanner s=new Scanner(System.in);
6         int r=s.nextInt(),c=s.nextInt();
7         int[][] A=new int[r][c],B=new int[r][c],R=new int[r][c];
8         for(int i=0;i<r;i++)for(int j=0;j<c;j++)A[i][j]=s.nextInt();
9         for(int i=0;i<r;i++)for(int j=0;j<c;j++)B[i][j]=s.nextInt();
10        for(;;){
11            int ch=s.nextInt(); if(ch==4)break;
12            if(ch<3)
13                for(int i=0;i<r;i++)for(int j=0;j<c;j++)
14                    R[i][j]=(ch==1)?A[i][j]+B[i][j]:A[i][j]-B[i][j];
15            else
16                for(int i=0;i<r;i++)for(int j=0;j<c;j++){
17                    R[i][j]=0;
18                    for(int k=0;k<c;k++)R[i][j]+=A[i][k]*B[k][j];
19                }
20                for(int[] row:R){for(int v:row)System.out.print(v+" ");System.out.println();}
21        }
22    }
23 }
```

Status: Runtime Error (NZEC)

Memory Used: 12312 byte | Execution Time: 0.181 sec

Exception in thread "main" java.util.NoSuchElementException
at java.base/java.util.Scanner.throwFor(Scanner.java:937)
at java.base/java.util.Scanner.next(Scanner.java:1594)
at java.base/java.util.Scanner.nextInt(Scanner.java:2258)
at java.base/java.util.Scanner.nextInt(Scanner.java:2212)
at Main.main(Main.java:6)

Input

Enter your input here...



J Access2.java 1 X

J Dog.class

J Access2.java > Language Support for Java(TM) by Red Hat > Access2

```
1 // Program 19 - Version 2
2 class Base {
3     public int pub = 5;
4     protected int pro = 6;
5     int def = 7;
6     private int pri = 8;
7     public int getPrivate() { return pri; }
8 }
9 class Child extends Base {
10    void printAll() {
11        System.out.println(pub + " " + pro + " " + def);
12        System.out.println("private = " + getPrivate());
13    }
14 }
15 public class Access2 {
16     Run main | Debug main | Run | Debug
17     public static void main(String[] a) {
18         new Child().printAll();
19     }
20 }
```

PROBLEMS 2

OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\" ;
5 6 7
private = 8

J Access1.java 1 X

J Dog.class

J Access1.java > ...

```
1 // Program 19 - Version 1
2 class Demo1 {
3     public int pub = 1;
4     protected int pro = 2;
5     int def = 3;
6     private int pri = 4;
7     void showPrivate() { System.out.println("private = " + pri); }
8 }
9
10 public class Access1 {
11     Run main | Debug main | Run | Debug
12     public static void main(String[] args) {
13         Demo1 d = new Demo1();
14         System.out.println(d.pub + " " + d.pro + " " + d.def);
15         d.showPrivate();
16     }
17 }
```

PROBLEMS 2

OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

● PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\" ;
1 2 3
private = 4

J AbstractDemo.java 1 X

J Dog.class

```
J AbstractDemo.java > ...
1 // Program 18 - Version 2
2 abstract class Animal {
3     public abstract void animalSound();
4     public void sleep() {
5         System.out.println("Zzz");
6     }
7 }
8 class Pig extends Animal {
9     public void animalSound() {
10        System.out.println("The pig says: wee wee");
11    }
12 }
13 public class AbstractDemo {
14     public static void main(String[] args) {
15         Pig myPig = new Pig();
16         myPig.animalSound();
17         myPig.sleep();
18     }
19 }
20
```

PROBLEMS 2

OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

- PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\" ;
The pig says: wee wee
Zzz

J AbstractDemo.java 2 X

J Dog.class

```
J AbstractDemo.java > ...
1 // Program 18 - Version 1
2 abstract class Shape {
3     abstract void draw(); // abstract method
4 }
5 class Rectangle extends Shape {
6     void draw() {
7         System.out.println(x:"drawing rectangle");
8     }
9 }
10 class Circle extends Shape {
11     void draw() {
12         System.out.println(x:"drawing circle");
13     }
14 }
15 public class AbstractDemo {
    Run main | Debug main | Run | Debug
16     public static void main(String args[]) {
17         Shape s = new Circle();
18         s.draw();
19         s = new Rectangle();
20         s.draw();
21     }
22 }
23
```

PROBLEMS 3

OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

```
PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\" ; i
drawing circle
drawing rectangle
```

J FinalizationDemo.java 5 X

J Dog.class

```
J FinalizationDemo.java > ...
1 // Program 17 - Version 2
2 class MyResource {
3     MyResource() {
4         System.out.println("Resource created.");
5     }
6     @Override
7     protected void finalize() {
8         System.out.println("Resource cleaned up via finalize().");
9     }
10 }
11 public class FinalizationDemo {
12     Run main | Debug main | Run | Debug
13     public static void main(String[] args) {
14         MyResource res = new MyResource();
15         res = null;
16         System.gc(); // Suggesting garbage collection
17         System.out.println("Main method is finishing.");
18     }
19 }
```

PROBLEMS 6 OUTPUT DEBUG CONSOLE TERMINAL PORTS

● PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\" ; javac FinalizationDemo.java
FinalizationDemo.java:7: warning: [removal] finalize() in Object has been deprecated and marked for removal.
protected void finalize() {
^
1 warning
Resource created.
Main method is finishing.
Resource cleaned up via finalize().

```
J FinalizationDemo.java 4 X J Dog.class
J FinalizationDemo.java > ...
1 // Program 17 - Version 1
2 public class FinalizationDemo {
3     Run main | Debug main | Run | Debug
4     public static void main(String[] args) {
5         FinalizationDemo obj = new FinalizationDemo();
6         System.out.println("Object hashcode: " + obj.hashCode());
7         obj = null;
8         // Requesting JVM to call Garbage Collector
9         System.gc();
10        System.out.println("End of the main method.");
11    }
12    @Override
13    protected void finalize() {
14        System.out.println("Finalize method called. The object is being collected.");
15    }
16 }
```

PROBLEMS 5 OUTPUT DEBUG CONSOLE TERMINAL PORTS

● PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\" ;
FinalizationDemo.java:12: warning: [removal] finalize() in Object has been deprecated and marked for removal.
protected void finalize() {
^
1 warning
Object hashcode: 622488023
End of the main method.
Finalize method called. The object is being collected.

```
J FinalKeywordDemo.java 2 X J Dog.class
J FinalKeywordDemo.java > ...
1 // Program 16 - Version 2
2 class Bike {
3     final void run() { // final method
4         System.out.println("running");
5     }
6 }
7 class Honda extends Bike {
8     // void run(){ System.out.println("running safely");} // Compile Error
9
10    final int speedlimit; // blank final variable
11
12    Honda() {
13        speedlimit = 70; // can be initialized only in constructor
14        System.out.println("Speed limit is: " + speedlimit);
15    }
16 }
17 public class FinalKeywordDemo {
18     Run main | Debug main | Run | Debug
19     public static void main(String args[]) {
20         new Honda();
21     }
22 }
```

PROBLEMS (3) OUTPUT DEBUG CONSOLE TERMINAL PORTS

● PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd

Speed limit is: 70

cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\" ;

PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM>

```
J FinalKeywordDemo.java X J Dog.class
J FinalKeywordDemo.java > ...
1 // Program 16 - Version 1
2 final class FinalClass { // final class
3     final int CONSTANT_VAR = 100; // final variable
4
5     final void finalMethod() { // final method
6         System.out.println("This is a final method.");
7         // CONSTANT_VAR = 200; // This would cause a compile error
8         System.out.println("Final variable value: " + CONSTANT_VAR);
9     }
10 }
11 // class SubClass extends FinalClass {} // Cannot inherit from final FinalClass
12 public class FinalKeywordDemo {
13     Run main | Debug main | Run | Debug
14     public static void main(String[] args) {
15         FinalClass obj = new FinalClass();
16         obj.finalMethod();
17     }
18 }
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS

● PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd

cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\" ;

This is a final method.
Final variable value: 100

J HierarchicalInheritance.java X

J Dog.class

```
J HierarchicalInheritance.java > ...
1 // Program 15 - Version 2
2 class Animal {
3     void makeSound() {
4         System.out.println("Animal makes a sound");
5     }
6 }
7 class Dog extends Animal { // Child 1
8     @Override
9     void makeSound() {
10        System.out.println("Dog barks");
11    }
12 }
13 class Cat extends Animal { // Child 2
14     @Override
15     void makeSound() {
16        System.out.println("Cat meows");
17    }
18 }
19 public class HierarchicalInheritance {
20     Run main | Debug main | Run | Debug
21     public static void main(String args[]) {
22         Animal myDog = new Dog();
23         Animal myCat = new Cat();
24         myDog.makeSound();
25         myCat.makeSound();
26     }
27 }
```

PROBLEMS 1

OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

- PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\" ;
nse }
Dog barks
● Cat meows

J MultilevelInheritance.java 3 X

J Dog.class

```
J MultilevelInheritance.java > ...
1 // Program 14 - Version 2
2 class Vehicle { // Grandparent
3     int speed = 50;
4 }
5 class Car extends Vehicle { // Parent
6     int speed = 100;
7 }
8 class SportsCar extends Car { // Child
9     int speed = 200;
10    void display() {
11        System.out.println("SportsCar speed: " + speed);
12        System.out.println("Car speed: " + super.speed); // super calls immediate parent
13    }
14 }
15 public class MultilevelInheritance {
16     Run main | Debug main | Run | Debug
17     public static void main(String args[]) {
18         SportsCar sc = new SportsCar();
19         sc.display();
20     }
21 }
```

PROBLEMS 4

OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

- PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\" ; java MultilevelInheritance
}
SportsCar speed: 200
Car speed: 100

J MultilevelInheritance.java X

```
J MultilevelInheritance.java > ...
1 // Program 14 - Version 1
2 class Animal { // Grandparent
3     void eat() {
4         System.out.println("eating...");
5     }
6 }
7 class Dog extends Animal { // Parent
8     void bark() {
9         System.out.println("barking...");
10    }
11 }
12 class BabyDog extends Dog { // Child
13     void weep() {
14         System.out.println("weeping...");
15     }
16     void show() {
17         super.eat(); // using super to call grandparent method
18         super.bark(); // using super to call parent method
19         weep();
20     }
21 }
22 public class MultilevelInheritance {
23     Run main | Debug main | Run | Debug
24     public static void main(String args[]) {
25         BabyDog d = new BabyDog();
26         d.show();
27     }
28 }
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS

- PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\" ;
}
eating...
● barking...
weeping...

J StaticDemo.java X

J StaticDemo.java > ...

```
1 // Program 13 - Version 2
2 ✓ class Student {
3     int rollno;
4     String name;
5     static String college = "ITS"; //static variable
6
7 ✓     Student(int r, String n) {
8         |    rollno = r;
9         |    name = n;
10    }
11
12 ✓     static void change(){ // static method
13         |    college = "BBDIT";
14    }
15
16 ✓     void display() {
17         |    System.out.println(rollno + " " + name + " " + college);
18    }
19 }
20 ✓ public class StaticDemo {
21     Run main | Debug main | Run | Debug
22     public static void main(String args[]) {
23         |    Student s1 = new Student(r:111, n:"Karan");
24         |    s1.display();
25         |    Student.change(); // calling change method
26         |    Student s2 = new Student(r:222, n:"Aryan");
27         |    s2.display();
28    }
29 }
```

PROBLEMS 1

OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

```
● PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\" ;
111 Karan ITS
222 Aryan BBDIT
```

J StaticDemo.java X

J StaticDemo.java > ...

```
1 // Program 13 - Version 1
2 class StaticExample {
3     static int a = 42; // static variable
4     static int b = 99; // static variable
5
6     static void callme() { // static method
7         System.out.println("a = " + a);
8     }
9 }
10 public class StaticDemo {
11     Run main | Debug main | Run | Debug
12     public static void main(String args[]) {
13         StaticExample.callme();
14         System.out.println("b = " + StaticExample.b);
15     }
16 }
```

PROBLEMS 1

OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

- PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\" ;

```
a = 42
b = 99
```

J OverloadingDemo.java X

```
J OverloadingDemo.java > ...
1 // Program 12 - Version 2
2 class Adder {
3     // Method overloading
4     static int add(int a, int b){return a+b;}
5     static double add(double a, double b){return a+b;}
6 }
7 class Box {
8     double width, height, depth;
9     // Constructor overloading
10    Box(double w, double h, double d) {
11        width = w; height = h; depth = d;
12    }
13    Box() {
14        width = height = depth = 0;
15    }
16    double volume() {
17        return width * height * depth;
18    }
19 }
20 public class OverloadingDemo {
21     Run main | Debug main | Run | Debug
22     public static void main(String[] args){
23         System.out.println(Adder.add(a:11,b:11));
24         System.out.println(Adder.add(a:12.3,b:12.6));
25
26         Box mybox1 = new Box(w:10, h:20, d:15);
27         Box mybox2 = new Box();
28         System.out.println("Volume of mybox1 is " + mybox1.volume());
29         System.out.println("Volume of mybox2 is " + mybox2.volume());
30     }
31 }
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS

● PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\" ;
22
24.9
Volume of mybox1 is 3000.0
Volume of mybox2 is 0.0

J OverloadingDemo.java 4 X

J OverloadingDemo.java > ...

```
1 // Program 12 - Version 1
2 class DisplayOverloading {
3     // Method overloading
4     public void disp(char c) {
5         System.out.println(c);
6     }
7     public void disp(char c, int num) {
8         System.out.println(c + " " + num);
9     }
10 }
11 class Sample {
12     // Constructor overloading
13     Sample() {
14         System.out.println("Default constructor called");
15     }
16     Sample(String str) {
17         System.out.println("Parameterized constructor called with: " + str);
18     }
19 }
20 public class OverloadingDemo {
21     Run main | Debug main | Run | Debug
22     public static void main(String args[]) {
23         DisplayOverloading obj = new DisplayOverloading();
24         obj.disp(c:'a');
25         obj.disp(c:'a',num:10);
26
27         Sample s1 = new Sample();
28         Sample s2 = new Sample(str:"Java");
29     }
30 }
```

PROBLEMS 5

OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

```
● PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\" ;
a
a 10
Default constructor called
Parameterized constructor called with: Java
```

J ConstructorDemo.java X

J ConstructorDemo.java > ...

```
1 // Program 11 - Version 2
2 class Student {
3     String name;
4     int id;
5
6     // Default constructor
7     Student() {
8         name = "Unknown";
9         id = 0;
10        System.out.println("Student object created with default values.");
11    }
12
13     // Parameterized constructor
14     Student(String n, int i) {
15         name = n;
16         id = i;
17         System.out.println("Student object created with given values.");
18     }
19
20     void getInfo() {
21         System.out.println("ID: " + id + ", Name: " + name);
22     }
23 }
24
25 public class ConstructorDemo {
26     public static void main(String[] args) {
27         Student s1 = new Student();
28         s1.getInfo();
29
30         Student s2 = new Student("Alex", 101);
31         s2.getInfo();
32     }
33 }
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
● PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\" ;
Student object created with default values.
ID: 0, Name: Unknown
Student object created with given values.
ID: 101, Name: Alex
```

J ConstructorDemo.java X

J ConstructorDemo.java > Language Support for Java(TM) by Red Hat > ConstructorDemo > main(String[])

```
1 // Program 11 - Version 1
2 class MyClass {
3     int num;
4
5     // Default constructor
6     MyClass() {
7         num = 100;
8         System.out.println("Default constructor called.");
9     }
10
11    // Parameterized constructor
12    MyClass(int n) {
13        num = n;
14        System.out.println("Parameterized constructor called.");
15    }
16
17    void display() {
18        System.out.println("Value of num: " + num);
19    }
20 }
21
22 public class ConstructorDemo {
23     Run main | Debug main | Run | Debug
24     public static void main(String[] args) {
25         MyClass obj1 = new MyClass();
26         obj1.display();
27
28         MyClass obj2 = new MyClass(n:200);
29         obj2.display();
30     }
31 }
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS

● PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\" ;
a ConstructorDemo }
Default constructor called.
Value of num: 100
Parameterized constructor called.
Value of num: 200

J Loops.java X

```
1 // Program 10 - Version 2: Different loop conditions
2 public class Loops {
3     Run | Debug | Run main | Debug main
4     public static void main(String[] args) {
5         System.out.println(x:"For Loop (descending):");
6         for(int i=3; i>=1; i--){
7             System.out.println(i);
8         }
9
10        System.out.println(x:"\nWhile Loop (stops at 1):");
11        int i=3;
12        while(i > 0){
13            System.out.println(i);
14            i--;
15        }
16
17        System.out.println(x:"\nDo-While Loop (runs once):");
18        int j=5;
19        do{
20            System.out.println(j);
21            j++;
22        } while(j < 5);
23    }
24 }
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS

● PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\" ;
For Loop (descending):

○ 3
2
1

While Loop (stops at 1):

3
2
1

Do-While Loop (runs once):

5

J Loops.java X

J Loops.java > ...

```
1 // Program 10 - Version 1: for, while, do-while
2 public class Loops {
3     Run | Debug | Run main | Debug main
4     public static void main(String[] args) {
5         System.out.println(x:"For Loop:");
6         for(int i=1; i<=3; i++){
7             System.out.println(i);
8         }
9         System.out.println(x:"\nWhile Loop:");
10        int i=1;
11        while(i<=3){
12            System.out.println(i);
13            i++;
14        }
15
16        System.out.println(x:"\nDo-While Loop:");
17        int j=1;
18        do{
19            System.out.println(j);
20            j++;
21        } while(j<=3);
22    }
23 }
24
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS

- PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\" ;
For Loop:
1
● 2
● 3
- While Loop:
1
● 2
3
- Do-While Loop:
1
2
3
- PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> []

J TernaryOperator.java X

J TernaryOperator.java > ...

```
1 // Program 9 - Version 2
2 public class TernaryOperator {
3     Run | Debug | Run main | Debug main
4     public static void main(String args[]) {
5         int number = 13;
6         String output = (number % 2 == 0) ? "Even number" : "Odd number";
7         System.out.println(output);
8     }
9 }
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS

● PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\" ; java TernaryOperator
}
Odd number
● PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM>

J TernaryOperator.java X

J TernaryOperator.java > ...

```
1 // Program 9 - Version 1
2 public class TernaryOperator {
3     Run | Debug | Run main | Debug main
4     public static void main(String args[]) {
5         int a = 20, b = 10;
6         String result = (a > b) ? "a is greater" : "b is greater";
7         System.out.println(result);
8     }
9 }
```

PROBLEMS 1

OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

- PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\" ;
}
a is greater

J SwitchStatement.java 1 X

J SwitchStatement.java > ...

```
2  public class SwitchStatement {  
    Run | Debug | Run main | Debug main  
3      public static void main(String[] args) {  
4          char grade = 'C';  
5          switch(grade) {  
6              case 'A' :  
7                  System.out.println("Excellent!");  
8                  break;  
9              case 'B' :  
10                 case 'C' :  
11                     System.out.println("Well done");  
12                     break;  
13                 default :  
14                     System.out.println("Invalid Grade");  
15             }  
16         }  
17     }  
18 }
```

PROBLEMS 2

OUTPUT

DEBUG CONSOLE

TERMINAL

POR

PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\" ; java SwitchStatement
Well done

J SwitchStatement.java 1 X

J SwitchStatement.java > ...

```
2  public class SwitchStatement {  
    Run | Debug | Run main | Debug main  
3      public static void main(String[] args) {  
4          int day = 4;  
5          String dayString;  
6          switch (day) {  
7              case 1: dayString = "Monday";  
8                  break;  
9              case 2: dayString = "Tuesday";  
10                 break;  
11              case 3: dayString = "Wednesday";  
12                  break;  
13              case 4: dayString = "Thursday";  
14                  break;  
15              default: dayString = "Invalid day";  
16                  break;  
17          }  
18          System.out.println(dayString);  
19      }  
20  }
```

PROBLEMS 2

OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

```
● PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\" ;  
}  
Thursday
```

J ConditionalStatements.java X

J ConditionalStatements.java > ...

```
1 // Program 7 - Version 2: if-else-if ladder
2 public class Conditionalstatements {
3     Run | Debug | Run main | Debug main
4     public static void main(String[] args) {
5         int marks = 78;
6         if (marks < 50) {
7             System.out.println("Fail");
8         } else if (marks >= 50 && marks < 60) {
9             System.out.println("D Grade");
10        } else if (marks >= 60 && marks < 70) {
11            System.out.println("C Grade");
12        } else if (marks >= 70 && marks < 80) {
13            System.out.println("B Grade");
14        } else {
15            System.out.println("A Grade");
16        }
17    }
18 }
```

PROBLEMS 1

OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

- PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\" ;
lstatements }
B Grade

J ConditionalStatements.java X

```
J ConditionalStatements.java > ...
1 // Program 7 - Version 1: if-else and nested if
2 public class ConditionalStatements {
3     Run | Debug | Run main | Debug main
4     public static void main(String[] args) {
5         int number = 10;
6         if (number > 0) {
7             System.out.println("The number is positive.");
8             if (number % 2 == 0) {
9                 System.out.println("And it is an even number.");
10            }
11        } else {
12            System.out.println("The number is not positive.");
13        }
14    }
15 }
```

PROBLEMS 1

OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

- PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\" ; :
lStatements }
The number is positive.
And it is an even number.

J JavaOperators.java

```
1 // Program 6 - Version 2
2 public class JavaOperators {
3     Run | Debug | Run main | Debug main
4     public static void main(String[] args) {
5         int a = 7; // 0111
6         int b = 4; // 0100
7         // Relational operator
8         System.out.println("a == b: " + (a == b)); // false
9         // Logical operator
10        System.out.println("a < 5 || b > 2: " + (a < 5 || b > 2)); // true
11        // Bitwise operator
12        System.out.println("a | b: " + (a | b)); // 7 (0111)
13    }
14
15
```

PROBLEMS 1

OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

```
● PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\" ;
a == b: false
a < 5 || b > 2: true
a | b: 7
```

J JavaOperators.java X

```
J JavaOperators.java > ...
1 // Program 6 - Version 1
2 public class JavaOperators {
3     Run | Debug | Run main | Debug main
4     public static void main(String[] args) {
5         int x = 10;
6         int y = 5;
7         // Relational operator
8         System.out.println("x > y: " + (x > y)); // true
9         // Logical operator
10        System.out.println("x > 5 && y < 10: " + (x > 5 && y < 10)); // true
11        // Bitwise operator
12        System.out.println("x & y: " + (x & y)); // 0
13    }
14 }
```

PROBLEMS 1

OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

- PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\" ;
x > y: true
x > 5 && y < 10: true
x & y: 0

J TypeCasting.java X

J TypeCasting.java > ...

```
1 // Program 5 - Version 2: Narrowing Casting (Explicit)
2 public class TypeCasting {
3     Run | Debug | Run main | Debug main
4     public static void main(String[] args) {
5         double myDouble = 9.78d;
6         int myInt = (int) myDouble; // Manual casting: double to int
7         System.out.println("Original double value: " + myDouble);
8         System.out.println("Casted int value: " + myInt);
9     }
10 }
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS

- PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\" ;
Original double value: 9.78
Casted int value: 9

J TypeCasting.java X

J TypeCasting.java > ...

```
1 // Program 5 - Version 1: Widening Casting (Implicit)
2 public class TypeCasting {
3     Run | Debug | Run main | Debug main
4     public static void main(String[] args) {
5         int myInt = 9;
6         double myDouble = myInt; // Automatic casting: int to double
7         System.out.println("Original int value: " + myInt);
8         System.out.println("Casted double value: " + myDouble);
9     }
10 }
```

PROBLEMS 1

OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

- PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\" ;
Original int value: 9
Casted double value: 9.0

J BasicDataTypes.java X

```
J BasicDataTypes.java > ...
1 // Program 4 - Version 2
2 public class BasicDataTypes {
3     Run | Debug | Run main | Debug main
4     public static void main(String[] args) {
5         byte myByte = 100;
6         short myShort = 5000;
7         long myLong = 15000000000L;
8         double myDouble = 19.99d;
9         System.out.println("Byte: " + myByte);
10        System.out.println("Short: " + myShort);
11        System.out.println("Long: " + myLong);
12        System.out.println("Double: " + myDouble);
13    }
14 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

- PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\" ;
Byte: 100
Short: 5000
- Long: 15000000000
Double: 19.99

J BasicDataTypes.java X

J AreaCalculator.class

J Room.class

J RoomAreaA.class

J BasicDataTypes.java > ...

```
1 // Program 4 - Version 1
2 public class BasicDataTypes {
3     Run | Debug | Run main | Debug main
4     public static void main(String[] args) {
5         int myNum = 5;                      // Integer (whole number)
6         float myFloatNum = 5.99f;           // Floating point number
7         char myLetter = 'D';               // Character
8         boolean myBool = true;             // Boolean
9         String myText = "Hello";          // String
10        System.out.println("Integer: " + myNum);
11        System.out.println("Float: " + myFloatNum);
12        System.out.println("Character: " + myLetter);
13        System.out.println("Boolean: " + myBool);
14        System.out.println("String: " + myText);
15    }
16 }
```

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

- PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\" ;
Integer: 5
Float: 5.99
- Character: D
Boolean: true
String: Hello

J CommandLineArguments.java X

J AreaCalculator.class

J Room.class

J RoomAreaA.class

```
J CommandLineArguments.java > ...
● 1 // Program 3 - Version 2
  2 public class CommandLineArguments {
    3     Run | Debug | Run main | Debug main
  4     public static void main(String[] args) {
  5         System.out.println("All command line arguments:");
  6         for (String arg : args) {
  7             System.out.println(arg);
  8         }
  9     }
10 }
```

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

```
● PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\"  
Arguments }  
All command line arguments:
```

J CommandLineArguments.java X

J AreaCalculator.class

J Room.class

J RoomAreaA.class

```
J CommandLineArguments.java > ...
1 // Program 3 - Version 1
2 public class CommandLineArguments {
3     Run | Debug | Run main | Debug main
4     public static void main(String[] args) {
5         if (args.length > 0) {
6             System.out.println("The first command line argument is: " + args[0]);
7         } else {
8             System.out.println("No command line arguments found.");
9         }
10    }
11 }
```

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

- PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\" ; if Arguments }
- No command line arguments found.

J RoomArea.java X

J AreaCalculator.class

J Room.class

J RoomAreaA.class

J RoomArea.java > ...

```
2  class Room {  
4      int breadth;  
5  
6      Room(int l, int b) {  
7          length = l;  
8          breadth = b;  
9      }  
10  
11     int calculateArea() {  
12         return length * breadth;  
13     }  
14 }  
15  
16 public class RoomArea {  
    Run | Debug | Run main | Debug main  
17     public static void main(String[] args) {  
18         Room room2 = new Room(l:15, b:25); // Changed dimensions  
19         System.out.println("The calculated area is: " + room2.calculateArea());  
20     }  
21 }  
22
```

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

● PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\" ;
The calculated area is: 375

J RoomAreaA.java X J AreaCalculator.class J Room.class J RoomAreaA.class

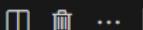


J RoomAreaA.java > Language Support for Java(TM) by Red Hat > AreaCalculator > computeArea(Room)

```
1 // File: RoomAreaA.java
2 class Room {
3     double length;
4     double width;
5     Room(double l, double w) {
6         length = l;
7         width = w;
8     }
9 }
10
11 class AreaCalculator {
12     double computeArea(Room r) {
13         return r.length * r.width;
14     }
15 }
16
17 public class RoomAreaA {
18     Run | Debug | Run main | Debug main
19     public static void main(String[] args) {
20         Room r = new Room(l:5.0, w:4.0);
21         AreaCalculator calc = new AreaCalculator();
22         System.out.println("Area = " + calc.computeArea(r));
23     }
24 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Code + ▾



```
● PS C:\Users\hello\Desktop\CODES\JAVA LAB - SAIM> cd "c:\Users\hello\Desktop\CODES\JAVA LAB - SAIM\" ; if ($?) { javac RoomAreaA.java } ; if ($?) { java RoomAreaA }
Area = 20.0
```

```
1 // Q1: WAP to print some statements like "Hello World!".
2 // Program 1 - Version 2
3 public class HelloWorld {
4     public static void main(String[] args) {
5         System.out.println("Hello, Universe!");
6     }
7 }
8 |
```

STDIN

Input for the program (Optional)

Output:

Hello, Universe!

```
1 // Q1: WAP to print some statements like "Hello World!".
2 // Program 1 - Version 1
3 public class HelloWorld {
4     public static void main(String[] args) {
5         System.out.println("Hello World!");
6     }
7 }
8
```

STDIN

Input for the program (Optional)

Output:

Hello World!