Q1. WAP to print some statements like “Hello World!”.

Code:

```
public class App {
    public static void main(String[] args) throws Exception {
        System.out.println("Hello, World!");
    }
}
```

```
src : vim — Konsole
public class App {
    public static void main(String[] args) throws Exception {
        System.out.println("Hello, World!");
    }
}
~
~
~
~
~
~
~
~
~
~
~
                                                    1,1           All
```

Output:

```
┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q1/src]
└─$ java App.java                                                    2 ⚙
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Hello, World!

┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q1/src]
└─$                                                                  2 ⚙
```

Q2. WAP to calculate room area using multiple classes.

Code:

```
import java.util.Scanner;

public class App {
   public static void main(String[] args) throws Exception {
      Scanner n = new Scanner(System.in);
      System.out.print("Enter the length of a Room :: ");
      double len =n.nextDouble();
      System.out.print("Enter the width of a Room :: ");
      double width =n.nextDouble();
      new Area(len, width);
   }
}
public class Area {
   Area(double length,double width){
      System.out.println("The area of a room is :: "+length*width);
   }
}
```

src : vim — Konsole

```
import java.util.Scanner;

public class App {
    public static void main(String[] args) throws Exception {
        Scanner n = new Scanner(System.in);
        System.out.print("Enter the length of a Room :: ");
        double len =n.nextDouble();
        System.out.print("Enter the width of a Room :: ");
        double width =n.nextDouble();
        new Area(len, width);
    }
}
~
```

src : vim — Konsole

```
public class Area {
    Area(double length,double width){
        System.out.println("The area of a room is :: "+length*width);
    }
}
~
```

output:

```
┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q2/bin]
└─$ java App                                                                    5 ⚙
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Enter the length of a Room :: 300
Enter the width of a Room :: 1500
The area of a room is :: 450000.0
```

Q3. WAP to demonstrate the use of command line arguments.

Code:

```
public class App {
   public static void main(String[] args) throws Exception {
      for(int i=0;i<args.length;i++){
         System.out.println(args[i]);
      }
   }
}
```

```
bin : vim — Konsole
public class App {
    public static void main(String[] args) throws Exception {
        for(int i=0;i<args.length;i++){
            System.out.println(args[i]);
        }
    }
}
~
```

Output:

```
bin : zsh — Konsole
┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q3/bin]
└─$ java App sahil_khan 21                                          6 ⚙
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
sahil_khan
21
```

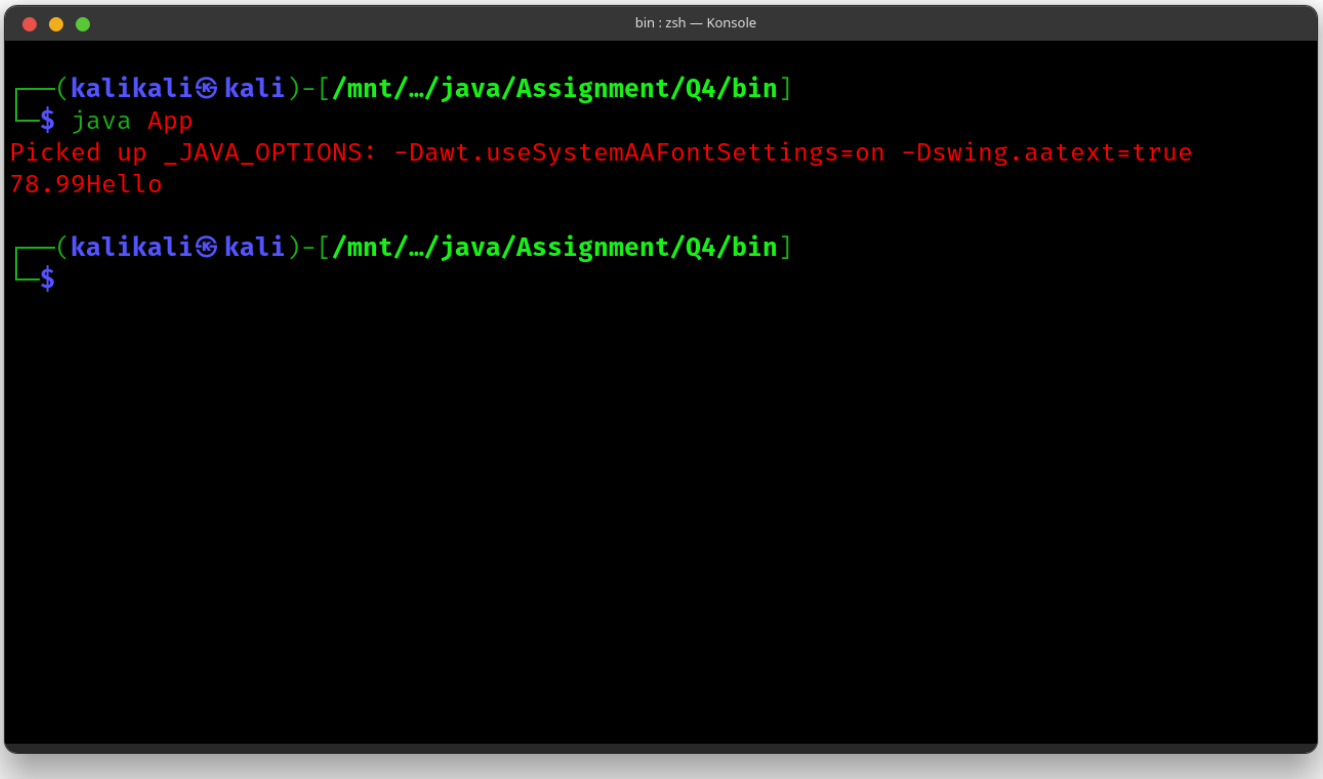
Q4. WAP to explain the basic data types used in java.

Code:

```
public class App {
    public static void main(String[] args) throws Exception {
        int myNum = 5; // Integer (whole number)
        float myFloatNum = 5.99f; // Floating point number
        char myLetter = 'D'; // Character
        boolean myBool = true; // Boolean
        String myText = "Hello"; // String
        if(myBool==true){
            System.out.println(myNum+myFloatNum+myLetter+myText);
        }
        else{
            System.out.println("Mybool is false");
        }
    }
}
```

```
src : vim — Konsole
public class App {
    public static void main(String[] args) throws Exception {
        int myNum = 5; // Integer (whole number)
        float myFloatNum = 5.99f; // Floating point number
        char myLetter = 'D'; // Character
        boolean myBool = true; // Boolean
        String myText = "Hello"; // String
        if(myBool==true){
            System.out.println(myNum+myFloatNum+myLetter+myText);
        }
        else{
            System.out.println("Mybool is false");
        }
    }
}
~
~
~
~
~
"App.java" 15L, 501B                                    1,1           All
```

Output:


bin : zsh — Konsole
┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q4/bin]
└─$ java App
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
78.99Hello
┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q4/bin]
└─$

Q5. WAP to explain the type casting in java.

Code:

```
public class explicitCasting {
    explicitCasting(){
        System.out.println("\n2. Explicit Casting");
        long longNum = 100;
        int intNum = (int) longNum;
        System.out.println(longNum +" is of type "+((Object)longNum).getClass().getSimpleName());
        System.out.println("Casting long to int!!");
        System.out.println(intNum +" is of type "+((Object)intNum).getClass().getSimpleName());
    }
}
public class implicitCasting {
    implicitCasting(){
        System.out.println("\n\n1.Implicit Casting ");
        byte n=100;
        int m =n;
        System.out.println(n +" is of type "+((Object)n).getClass().getSimpleName());
        System.out.println("Casting byte to int!!");
        System.out.println(m +" is of type "+((Object)m).getClass().getSimpleName());
    }
}
public class App {
    public static void main(String[] args) throws Exception {
        System.out.println("There are Two types of Casting \n1. Implicit Casting\n2. Explicit Casting");
        new implicitCasting();
        new explicitCasting();
    }
}
```

src : vim — Konsole

```
public class App {
    public static void main(String[] args) throws Exception {
        System.out.println("There are Two types of Casting \n1. Implicit Casting\n2.
 Explicit Casting");
        new implicitCasting();
        new explicitCasting();
    }
}
~
~
~
~
~
~
~
~
~
~
~
~
~
"App.java" 7L, 256B                                          1,1           All
```

src : vim — Konsole

```
public class implicitCasting {
    implicitCasting(){
        System.out.println("\n\n1.Implicit Casting ");
        byte n=100;
        int m =n;
        System.out.println(n +" is of type "+((Object)n).getClass().getSimpleName());
        System.out.println("Casting byte to int!!");
        System.out.println(m +" is of type "+((Object)m).getClass().getSimpleName());
    }
}
~
~
~
~
~
~
~
~
~
~
~
                                                             1,1           All
```

src : vim — Konsole

```
public class explicitCasting {
    explicitCasting(){
        System.out.println("\n2. Explicit Casting");
        long longNum = 100;
        int intNum = (int) longNum;
        System.out.println(longNum +" is of type "+((Object)longNum).getClass().getSimpleName());
        System.out.println("Casting long to int!!");
        System.out.println(intNum +" is of type "+((Object)intNum).getClass().getSimpleName());
    }
}
~
~
~
~
~
~
~
~
~
~
~
~
~
~
                                                                    1,1           All
```

Output:

```
┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q5/src]
└─$ java App                                                                3 ⚙
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
There are Two types of Casting
1. Implicit Casting
2. Explicit Casting


1.Implicit Casting
100 is of type Byte
Casting byte to int!!
100 is of type Integer
2. Explicit Casting
100 is of type Long
Casting long to int!!
100 is of type Integer

┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q5/src]
└─$                                                                         3 ⚙
```

Q6. WAP to demonstrate java expressions using different operators in java like relational, logical, bitwise operators etc.

Code:

```
import java.util.Scanner;
public class RelationalOperators {
   RelationalOperators(){
      System.out.println("\n\n1.Rational  Condition");
      System.out.println("Enter  the value of i :: ");
      Scanner n = new Scanner(System.in);
      int i=n.nextInt();
      if( i==0){
         System.out.println("i  is 0");
      }
      else if(i==1){
         System.out.println("i  is 1");
      }
      else{
         System.out.println("All  of above rational condition  are false");
      }
   }
}
public class logicalOperators {
   logicalOperators(){
      System.out.println("\n\n1.Logical  Condition");
      int a=50,b=30,c=100;
      boolean condition = true;

      // AND LOGICAL OPERATOR
      if(a<c && b<c){
         System.out.println(c+"  is greatest  number");
      }

      // OR LOGICAL OPERATOR
      else if(a%2==0 || b%2==0 || c%2==0){
         System.out.println("one  of them even number");
      }

      // NOT LOGICAL OPERATOR
      else if(!condition){
         System.out.println("Condition  is false!!");
      }

      else{
         System.out.println("Logical  operators");
      }
   }
```

```
}
public class BitwiseOperator {
   BitwiseOperator() {
      System.out.println("\n\n1.Bitwise  Condition");
      int x = 9, y = 8;

      // bitwise and
      // 1001 & 1000 = 1000 = 8
      System.out.println("x  & y = " + (x & y));

      // bitwise XOR
      // 1001 ^ 1000 = 0001 = 1
      System.out.println("x  ^ y = " + (x ^ y));

      // bitwise inclusive  OR
      // 1001 | 1000 =  1001 = 9
      System.out.println("x  | y = " + (x | y));

      // bitwise compliment
      // ~0010= 1101 = -3
      System.out.println("~x  = " + (~x));
   }
}
public class App {
   public static void main(String[] args) throws Exception {
      System.out.println("WAP  to demonstrate java expressions using different operators in java like
relational, logical, bitwise operators etc.");
      new RelationalOperators();
      new logicalOperators();
      new BitwiseOperator();
   }
}
```

src : vim — Konsole

```
public class App {
    public static void main(String[] args) throws Exception {
        System.out.println("WAP to demonstrate java expressions using different oper
ators in java like relational, logical, bitwise operators etc.");
        new RelationalOperators();
        new logicalOperators();
        new BitwiseOperator();
    }
}
~
~
~
~
~
~
~
~
~
~
~
"App.java" 8L, 337B                                          1,1           All
```

src : vim — Konsole

```
import java.util.Scanner;
public class RelationalOperators {
    RelationalOperators(){
        System.out.println("\n\n1.Rational Condition");
        System.out.println("Enter the value of i :: ");
        Scanner n = new Scanner(System.in);
        int i=n.nextInt();
        if( i==0){
            System.out.println("i is 0");
        }
        else if(i==1){
            System.out.println("i is 1");
        }
        else{
            System.out.println("All of above rational condition are false");
        }
    }
}
~
~
"RelationalOperators.java" 18L, 526B                         1,1           All
```

src : vim — Konsole

```
public class logicalOperators {
    logicalOperators(){
        System.out.println("\n\n1.Logical Condition");
        int a=50,b=30,c=100;
        boolean condition = true;

        // AND LOGICAL OPERATOR
        if(a<c && b<c){
            System.out.println(c+" is greatest number");
        }

        // OR LOGICAL OPERATOR
        else if(a%2==0 || b%2==0 || c%2==0){
            System.out.println("one of them even number");
        }

        // NOT LOGICAL OPERATOR
        else if(!condition){
            System.out.println("Condition is false!!");
        }
                                                        15,1          Top
```

src : vim — Konsole

```
        // NOT LOGICAL OPERATOR
        else if(!condition){
            System.out.println("Condition is false!!");
        }

        else{
            System.out.println("Logical operators");
        }
    }
}
~
~
~
~
~
~
~
~
~
                                                        24,9          Bot
```

src : vim — Konsole

```java
public class BitwiseOperator {
    BitwiseOperator() {
        System.out.println("\n\n1.Bitwise Condition");
        int x = 9, y = 8;

        // bitwise and
        // 1001 & 1000 = 1000 = 8
        System.out.println("x & y = " + (x & y));

        // bitwise XOR
        // 1001 ^ 1000 = 0001 = 1
        System.out.println("x ^ y = " + (x ^ y));

        // bitwise inclusive OR
        // 1001 | 1000 = 1001 = 9
        System.out.println("x | y = " + (x | y));

        // bitwise compliment
        // ~0010= 1101 = -3
        System.out.println("~x = " + (~x));
```

15,1 Top

src : vim — Konsole

```java
        // bitwise XOR
        // 1001 ^ 1000 = 0001 = 1
        System.out.println("x ^ y = " + (x ^ y));

        // bitwise inclusive OR
        // 1001 | 1000 = 1001 = 9
        System.out.println("x | y = " + (x | y));

        // bitwise compliment
        // ~0010= 1101 = -3
        System.out.println("~x = " + (~x));
    }
}
~
~
~
~
~
~
~
```

15,33 Bot

Output:

```
bin : zsh — Konsole
└─$ java App                                                                    7 ⚙
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
WAP to demonstrate java expressions using different operators in java like relationa
l, logical, bitwise operators etc.


1.Rational Condition
Enter the value of i ::
0
i is 0


1.Logical Condition
100 is greatest number


1.Bitwise Condition
x & y = 8
x ^ y = 1
x | y = 9
~x = -10
```

Q7. WAP to demonstrate if-else, nested if-else, if-else ladder.

Code:

```
import java.util.Scanner;

public class ifElse {
    ifElse(){
        System.out.println("\n\n1. if else");
        System.out.print("Enter the number :: ");
        Scanner n = new Scanner(System.in);
        int num=n.nextInt();
        if(num%2==0){
            System.out.println(num+" is a even number");
        }
        else{
            System.out.println(num+" is a odd number");
        }
    }
}
import java.util.Scanner;

public class nestedIfelse {
    nestedIfelse(){
        System.out.println("\n\n1. Nested if else");
        System.out.print("Enter the number :: ");
        Scanner n = new Scanner(System.in);
        int num=n.nextInt();
        if(num<100){
            if(num%2==0){
                System.out.println(num+" is smaller than 100 and even number");
            }
            else{
                System.out.println(num+" is smaller than 100 but not even number (odd number)");
            }
        }
        else{
            System.out.println(num+" is greater than 100");
        }
    }
}
import java.util.Scanner;

public class ladderIfelse {
    ladderIfelse(){
```

```
        System.out.println("\n\n1. Nested if else");
        System.out.print("Enter the number :: ");
        Scanner n = new Scanner(System.in);
        int num=n.nextInt();
        if (num<0) {
            System.out.println(num+" is a negative number");
        }
        else if (num==0){
            System.out.println(num+" is zero");
        }
        else if(num>0){
            System.out.println(num+" is a positive number");
        }
        else{
            System.out.println("invalid input!!");
        }
    }
}
public class App {
    public static void main(String[] args) throws Exception {
        System.out.println("WAP to demonstrate if-else, nested if-else, if-else ladder.");
        new ifElse();
        new nestedIfelse();
        new ladderIfelse();
    }
}
```

src : vim — Konsole

```java
public class App {
    public static void main(String[] args) throws Exception {
        System.out.println("WAP to demonstrate if-else, nested if-else, if-else ladder.");
        new ifElse();
        new nestedIfelse();
        new ladderIfelse();
    }
}
~
~
~
~
~
~
~
~
~
~
~
"App.java" 8L, 258B                                    1,1           All
```

src : vim — Konsole

```java
import java.util.Scanner;

public class ifElse {
    ifElse(){
        System.out.println("\n\n1. if else");
        System.out.print("Enter the number :: ");
        Scanner n = new Scanner(System.in);
        int num=n.nextInt();
        if(num%2==0){
            System.out.println(num+" is a even number");
        }
        else{
            System.out.println(num+" is a odd number");
        }
    }
}
~
~
~
~
"ifElse.java" 16L, 409B                                1,1           All
```

src : vim — Konsole

```
import java.util.Scanner;

public class nestedIfelse {
    nestedIfelse(){
        System.out.println("\n\n1. Nested if else");
        System.out.print("Enter the number :: ");
        Scanner n = new Scanner(System.in);
        int num=n.nextInt();
        if(num<100){
            if(num%2==0){
                System.out.println(num+" is smaller than 100 and even number");
            }
            else{
                System.out.println(num+" is smaller than 100 but not even number (od
d number)");
            }
        }
        else{
            System.out.println(num+" is greater than 100");
        }
                                                                    14,1          Top
```

src : vim — Konsole

```
            if(num%2==0){
                System.out.println(num+" is smaller than 100 and even number");
            }
            else{
                System.out.println(num+" is smaller than 100 but not even number (od
d number)");
            }
        }
        else{
            System.out.println(num+" is greater than 100");
        }
    }
}
~
~
~
~
~
~
~
                                                                    15,13         Bot
```

src : vim — Konsole

```java
import java.util.Scanner;

public class ladderIfelse {
    ladderIfelse(){
        System.out.println("\n\n1. Nested if else");
        System.out.print("Enter the number :: ");
        Scanner n = new Scanner(System.in);
        int num=n.nextInt();
        if (num<0) {
            System.out.println(num+" is a negative number");
        }
        else if (num==0){
            System.out.println(num+" is zero");
        }
        else if(num>0){
            System.out.println(num+" is a positive number");
        }
        else{
            System.out.println("invalid input!!");
        }
                                                                15,1          Top
```

src : vim — Konsole

```java
        else if(num>0){
            System.out.println(num+" is a positive number");
        }
        else{
            System.out.println("invalid input!!");
        }
    }
}
~
~
~
~
~
~
~
~
~
~
~
~
                                                                20,9          Bot
```

Output:

```
bin : zsh — Konsole
└─$ java App                                                                    11 ⚙
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
WAP to demonstrate if-else, nested if-else, if-else ladder.


1. if else
Enter the number :: 10
10 is a even number


1. Nested if else
Enter the number :: 12
12 is smaller than 100 and even number


1. Nested if else
Enter the number :: 4
4 is a positive number

┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q7/bin]
└─$                                                                             11 ⚙
```

Q8. WAP to demonstrate switch statements.

Code:

```
import java.util.Scanner;

public class switchCase {
    switchCase(){
        System.out.println("\n\nSwitch Case");
        System.out.print("Enter the day number (less than 8) ::");
        Scanner n = new Scanner(System.in);
        int dayNum = n.nextInt();
        String dayName;
        switch (dayNum) {
            case 1:
                dayName = "Monday";
                break;

            case 2:
                dayName = "Tuesday";
                break;

            case 3:
                dayName = "Wednesday";
                break;

            case 4:
                dayName = "Thursday";
                break;

            case 5:
                dayName = "Friday";
                break;

            case 6:
                dayName = "Saturday";
                break;

            case 7:
                dayName = "Sunday";
                break;

            default:
                dayName = "Invalid Day";
                break;
```

```
        }
        System.out.println(dayName);
    }
}
public class App {
    public static void main(String[] args) throws Exception {
        System.out.println("Switch case!!");
        new switchCase();
    }
}
```

src : vim — Konsole

```
public class App {
    public static void main(String[] args) throws Exception {
        System.out.println("Switch case!! ");
        new switchCase();
    }
}
~
~
~
~
~
~
~
~
~
~
~
~
~
~
"App.java" 6L, 160B                                    1,1           All
```

src : vim — Konsole

```
import java.util.Scanner;

public class switchCase {
    switchCase(){
        System.out.println("\n\nSwitch Case");
        System.out.print("Enter the day number (less than 8) ::");
        Scanner n = new Scanner(System.in);
        int dayNum = n.nextInt();
        String dayName;
        switch (dayNum) {
            case 1:
                dayName = "Monday";
                break;

            case 2:
                dayName = "Tuesday";
                break;

            case 3:
                dayName = "Wednesday";
                                                       12,1          Top
```

src : vim — Konsole

```
            case 3:
                dayName = "Wednesday";
                break;

            case 4:
                dayName = "Thursday";
                break;

            case 5:
                dayName = "Friday";
                break;

            case 6:
                dayName = "Saturday";
                break;

            case 7:
                dayName = "Sunday";
                break;
                                                    26,8          68%
```

src : vim — Konsole

```
                dayName = "Saturday";
                break;

            case 7:
                dayName = "Sunday";
                break;

            default:
                dayName = "Invalid Day";
                break;
        }
        System.out.println(dayName);
    }
}
~
~
~
~
~
~
                                                    40,40         Bot
```

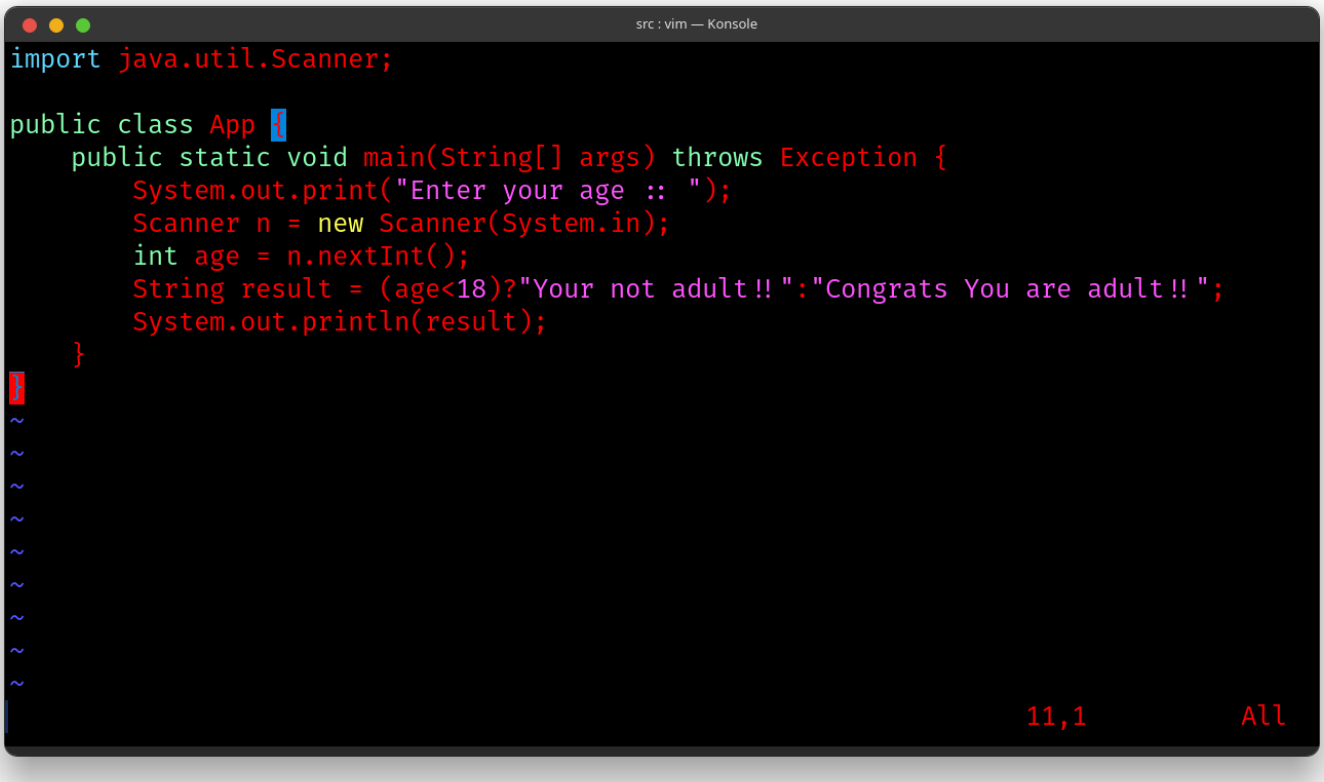
Output:

```
bin : zsh — Konsole

┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q8/bin]
└─$ java App                                                          13 ⚙
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Switch case!!


Switch Case
Enter the day number (less than 8) ::7
Sunday

┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q8/bin]
└─$                                                                   13 ⚙
```

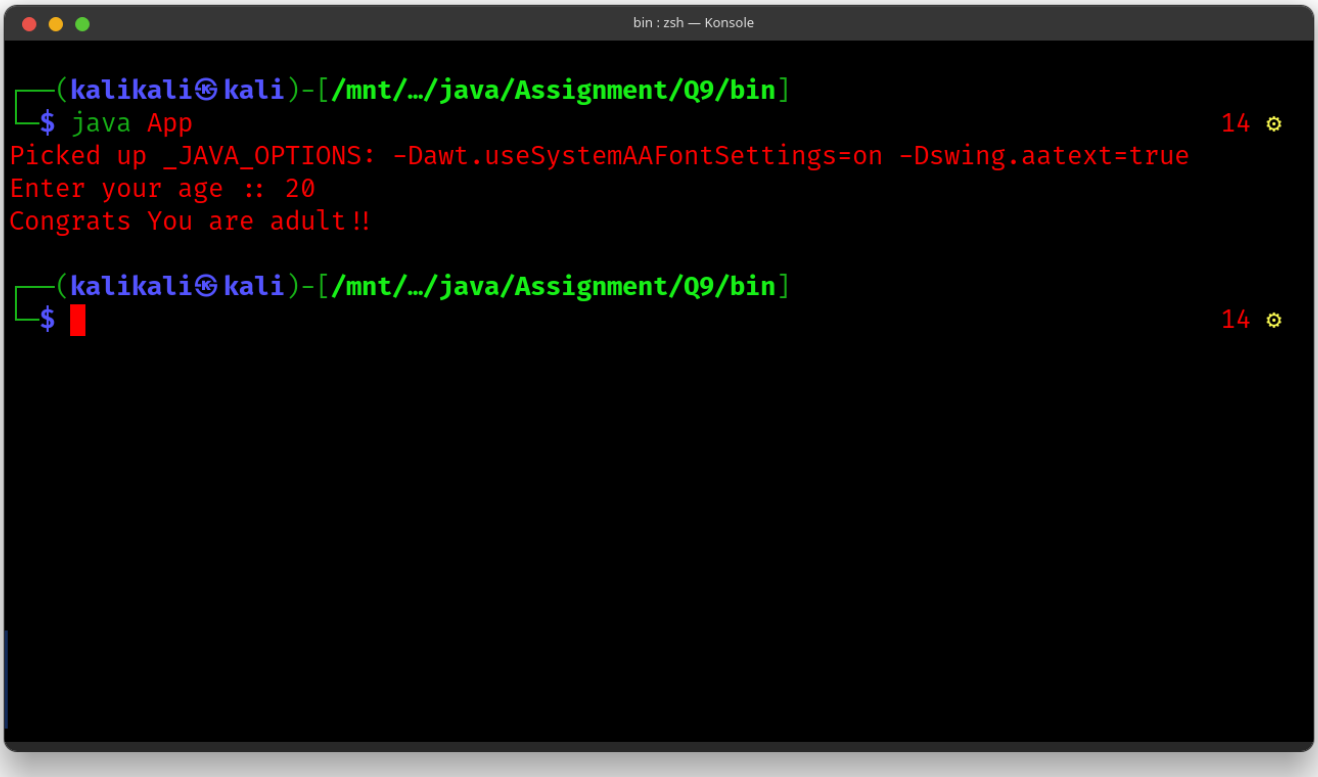
Q9. WAP to demonstrate "? :" operator.

Code:

```
import java.util.Scanner;

public class App {
   public static void main(String[] args) throws Exception {
      System.out.print("Enter your age :: ");
      Scanner n = new Scanner(System.in);
      int age = n.nextInt();
      String result = (age<18)?"Your not adult!!":"Congrats You are adult!!";
      System.out.println(result);
   }
}
```



Output:


bin : zsh — Konsole
┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q9/bin]
└─$ java App
14
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Enter your age :: 20
Congrats You are adult!!
┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q9/bin]
└─$
14

Q10. WAP to explain the working of while, for and do-while loops.

Code:

```
public class forLoop {
   forLoop(){
      int i;
      System.out.println("\n1. For Loop!!");
      for(i=0;i<100;i++){
         System.out.println(i);
      }
   }
}
public class whileLoop {
   whileLoop(){
      System.out.println("\n\n2. While Loop!!");
      int i =0;
      while (i<100) {
         System.out.println(i);
         i++;
      }
   }
}
public class doWhileLoop {
   doWhileLoop(){
      System.out.println("\n\n3. Do While Loop!!");
      int i =0;
      do {
         System.out.println(i);
         i++;
      } while (i<100);
   }
}
public class App {
   public static void main(String[] args) throws Exception {
      new forLoop();
      new whileLoop();
      new doWhileLoop();
   }
```

```
}
```

src : vim — Konsole

```
public class App {
    public static void main(String[] args) throws Exception {
        new forLoop();
        new whileLoop();
        new doWhileLoop();
    }
}
~
~
~
~
~
~
~
~
~
~
~
~
~
"App.java" 7L, 166B                                    1,1           All
```

src : vim — Konsole

```
public class forLoop {
    forLoop(){
        int i;
        System.out.println("\n1. For Loop!!");
        for(i=0;i<100;i++){
            System.out.println(i);
        }
    }
}
~
~
~
~
~
~
~
~
~
~
~
"forLoop.java" 9L, 181B                                1,1           All
```

src : vim — Konsole

```java
public class whileLoop {
    whileLoop(){
        System.out.println("\n\n2. While Loop!!");
        int i =0;
        while (i<100) {
            System.out.println(i);
            i++;
        }
    }
}
~
~
~
~
~
~
~
~
~
~
"whileLoop.java" 10L, 205B                                  1,1           All
```

src : vim — Konsole

```java
public class doWhileLoop {
    doWhileLoop(){
        System.out.println("\n\n3. Do While Loop!!");
        int i =0;
        do {
            System.out.println(i);
            i++;
        } while (i<100);
    }
}
~
~
~
~
~
~
~
~
~
~
"doWhileLoop.java" 10L, 216B                                1,1           All
```

Output:

```
1. For Loop!!
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
```

```
98
99


2. While Loop!!
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
```

```
97
98
99


3. Do While Loop!!
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
```

Q11. WAP to demonstrate different types of constructors in java like default and parameterized.

Code:

```
public class Student {
    Student(){
        System.out.println("like default!!");
    }
    Student(String name, int age){
        System.out.println("name :: "+name+"\nage :: "+age);
    }
}
public class App {
    public static void main(String[] args) throws Exception {
        new Student();
        new Student("Shakshi", 21);
    }
}
```

src : vim — Konsole

```
public class App {
    public static void main(String[] args) throws Exception {
        new Student();
        new Student("Sahil khan", 21);
    }
}
~
~
~
~
~
~
~
~
~
~
~
~
~
~
"App.java" 6L, 151B                                    1,1           All
```

src : vim — Konsole

```
public class Student {
    Student(){
        System.out.println("like default!!");
    }
    Student(String name, int age){
        System.out.println("name :: "+name+"\nage :: "+age);
    }
}
~
~
~
~
~
~
~
~
~
~
~
~
                                                       8,1           All
```

Output:

```
bin : zsh — Konsole

┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q11/bin]
└─$ java App                                                          22 ⚙
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
like default!!
name :: Sahil khan
age :: 21

┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q11/bin]
└─$                                                                   22 ⚙
```

Q12. WAP to explain method overloading and constructor overloading.

Code:

```
public class Add {
   Add(){
      System.out.println("\n\n1. Consrutor Overloading!!");
   }
   Add(int a, int b){
      System.out.println("The sum of "+a+" and "+b+" is :: "+(a+b));
   }
   Add(double a, double b){
      System.out.println("The sum of "+a+" and "+b+" is :: "+(a+b));
   }
}
public class Sub {
   Sub(){
      System.out.println("\n\n2. Method Overloading!!");
   }
   void subtract(int a,int b){
      System.out.println("The subtraction :: "+(a-b));
   }
   void subtract(double a, double b){
      System.out.println("The subtraction :: "+(a-b));
   }
}
public class App {
   public static void main(String[] args) throws Exception {
      System.out.println("Method overloading and constructor overloading!!");
      new Add();
      new Add(5, 10);
      new Add(5.5, 10.5);
      Sub num = new Sub();
      num.subtract(10, 5);
      num.subtract(100.5, 200.4);
   }
}
```

src : vim — Konsole

```java
public class App {
    public static void main(String[] args) throws Exception {
        System.out.println("Method overloading and constructor overloading!!");
        new Add();
        new Add(5, 10);
        new Add(5.5, 10.5);
        Sub num = new Sub();
        num.subtract(10, 5);
        num.subtract(100.5, 200.4);
    }
}
~
~
~
~
~
~
~
~
~
"App.java" 11L, 334B                                    1,1           All
```

src : vim — Konsole

```java
public class Add {
    Add(){
        System.out.println("\n\n1. Consrutor Overloading!!");
    }
    Add(int a, int b){
        System.out.println("The sum of "+a+" and "+b+" is :: "+(a+b));
    }
    Add(double a, double b){
        System.out.println("The sum of "+a+" and "+b+" is :: "+(a+b));
    }
}
~
~
~
~
~
~
~
~
~
"Add.java" 11L, 306B                                    1,1           All
```

src : vim — Konsole

```
public class Sub {
    Sub(){
        System.out.println("\n\n2. Method Overloading!!");
    }
    void subtract(int a,int b){
        System.out.println("The subtraction :: "+(a-b));
    }
    void subtract(double a, double b){
        System.out.println("The subtraction :: "+(a-b));
    }
}
~
~
~
~
~
~
~
~
~
"Sub.java" 11L, 294B                                    1,1           All
```

Output:

```
┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q12/bin]
└─$ java App                                                          25 ⚙
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Method overloading and constructor overloading!!


1. Consrutor Overloading!!
The sum of 5 and 10 is :: 15
The sum of 5.5 and 10.5 is :: 16.0


2. Method Overloading!!
The subtraction :: 5
The subtraction :: -99.9

┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q12/bin]
└─$                                                                   25 ⚙
```

Q13. WAP to explain static methods and static members.

Code:

```
public class App {
    // static variable
    static int a = 40;

    // instance variable
    int b = 50;

    void simpleDisplay() {
        System.out.println("\n\n1. Simple Display Method\n" + a);
        System.out.println(b);
    }

    // Declaration of a static method.
    static void staticDisplay() {
        System.out.println("\n\n2. Static Display Method\n" + a);
    }

    public static void main(String[] args) throws Exception {
        System.out.println("Static variable & Static Method!!");
        App obj = new App();

        // calling simple method
        obj.simpleDisplay();

        // calling static method
        staticDisplay();
    }
}
```

src : vim — Konsole

```java
public class App {
    // static variable
    static int a = 40;

    // instance variable
    int b = 50;

    void simpleDisplay() {
        System.out.println("\n\n1. Simple Display Method\n" + a);
        System.out.println(b);
    }

    // Declaration of a static method.
    static void staticDisplay() {
        System.out.println("\n\n2. Static Display Method\n" + a);
    }

    public static void main(String[] args) throws Exception {
        System.out.println("Static variable & Static Method!!");
        App obj = new App();
                                                                15,1          Top
```

src : vim — Konsole

```java
    }

    public static void main(String[] args) throws Exception {
        System.out.println("Static variable & Static Method!!");
        App obj = new App();

        // calling simple method
        obj.simpleDisplay();

        // calling static method
        staticDisplay();
    }
}
~
~
~
~
~
~
~
                                                                21,0-1        Bot
```

Output:

```
┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q13/bin]
└─$ java App                                                          26 ⚙
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Static variable & Static Method!!


1. Simple Display Method
40
50


2. Static Display Method
40

┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q13/bin]
└─$                                                                   26 ⚙
```

Q14. WAP to demonstrate multilevel inheritance and super keyword.

Code:

```
class Car {
   String warranty ="10 yr";
   Car(){
      System.out.println("Class Car");
   }
   void vehicleType(){
      System.out.println("Vehicle Type : 4 wheeler");
   }
}
class Maruti extends Car{
   Maruti(){
      System.out.println("Class Maruti");
   }
   void brand(){
      System.out.println("Brand : Maruti");
   }
   void speed(){
      System.out.println("Max : 200kmph");
   }
   void warranty(){
      System.out.println("Warranty : "+super.warranty);
   }
}
class Maruti800 extends Maruti{
   Maruti800(){
      System.out.println("Maruti Model : 800");
   }
   void speed(){
      System.out.println("Max : 100kmph");
   }
   void warranty(){
      System.out.println("Warranty : "+super.warranty);
   }
}
public class App {
```

```
    public static void main(String[] args) throws Exception {
        Maruti800 car1 = new Maruti800();
        car1.vehicleType();
        car1.warranty();
        car1.speed();
    }
}
```

src : vim — Konsole

```
public class App {
    public static void main(String[] args) throws Exception {
        Maruti800 car1 = new Maruti800();
        car1.vehicleType();
        car1.warranty();
        car1.speed();
    }
}
~
~
~
~
~
~
~
~
~
~
~
"App.java" 8L, 206B                                    1,1           All
```

src : vim — Konsole

```
class Car {
    String warranty ="10 yr";
    Car(){
        System.out.println("Class Car");
    }
    void vehicleType(){
        System.out.println("Vehicle Type : 4 wheeler");
    }
}
class Maruti extends Car{
    Maruti(){
        System.out.println("Class Maruti");
    }
    void brand(){
        System.out.println("Brand : Maruti");
    }
    void speed(){
        System.out.println("Max : 200kmph");
    }
    void warranty(){
                                                           15,1          Top
```

```
    }
    void warranty(){
        System.out.println("Warranty : "+super.warranty);
    }
}
class Maruti800 extends Maruti{
    Maruti800(){
        System.out.println("Maruti Model : 800");
    }
    void speed(){
        System.out.println("Max : 100kmph");
    }
    void warranty(){
        System.out.println("Warranty : "+super.warranty);
    }
}
~
~
~
~
                                                    24,31         Bot
```

Output:

```
┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q14/bin]
└─$ java App                                                          28 ⚙
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Class Car
Class Maruti
Maruti Model : 800
Vehicle Type : 4 wheeler
Warranty : 10 yr
Max : 100kmph

┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q14/bin]
└─$                                                                   28 ⚙
```

Q15. WAP to demonstrate method overriding in hierarchical inheritance.

Code:

```
public class Hierarchical {
    void printInfo(){
        System.out.println("Hierarchical class");
    }
}
class A extends Hierarchical{
    void printInfo(){
        System.out.println("A class");
    }
}
class B extends Hierarchical{
    void printInfo(){
        System.out.println("B class");
    }
}
class C extends Hierarchical{
    void printInfo(){
        System.out.println("C class");
    }
}
public class App {
    public static void main(String[] args) throws Exception {
        System.out.println("Q. WAP to demonstrate method overriding in hierarchical
inheritance.\n\n");
        A objA = new A();
        objA.printInfo();

        B objB = new B();
        objB.printInfo();


        C objC = new C();
        objC.printInfo();
    }
}
```

src : vim — Konsole

```java
public class App {
    public static void main(String[] args) throws Exception {
        System.out.println("Q. WAP to demonstrate method overriding in hierarchical
inheritance.\n\n");
        A objA = new A();
        objA.printInfo();

        B objB = new B();
        objB.printInfo();


        C objC = new C();
        objC.printInfo();
    }
}
~
~
~
~
~
"App.java" 14L, 360B                                          1,1           All
```

src : vim — Konsole

```java
public class Hierarchical {
    void printInfo(){
        System.out.println("Hierarchical class");
    }
}
class A extends Hierarchical{
    void printInfo(){
        System.out.println("A class");
    }
}
class B extends Hierarchical{
    void printInfo(){
        System.out.println("B class");
    }
}
class C extends Hierarchical{
    void printInfo(){
        System.out.println("C class");
    }
}
                                                              20,1          All
```

Output:

```
┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q15/bin]
└─$ java App                                                          30 ⚙
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Q. WAP to demonstrate method overriding in hierarchical inheritance.


A class
B class
C class

┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q15/bin]
└─$                                                                   30 ⚙
```

Q16. WAP to explain the working of final classes and also use final methods and final variables in your program.

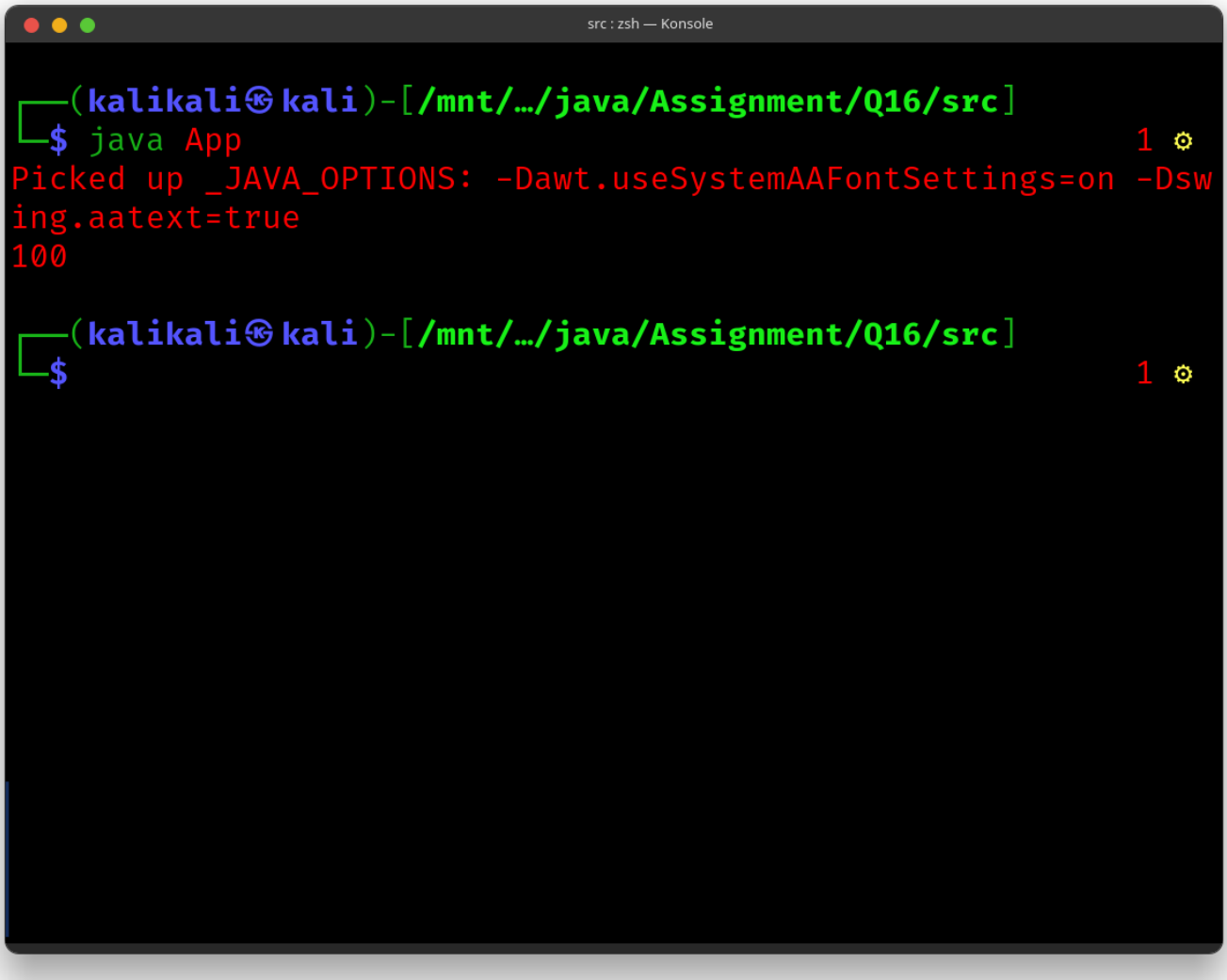
Code:

```
public class App {
    public static void main(String[] args) throws Exception {
        finalClass obj = new finalClass();
        obj.finalMethod();
    }
}

// final keyword means we cant overwrite the value of any final object in future during writing code
other wise it gives an error "The final local variable finalVariable cannot be assigned. It must be blank
and not using a compound" assignment
final class finalClass{
    final void finalMethod(){
        final int finalVariable=100;
        System.out.println(finalVariable);
    }
}
```

src : vim — Konsole

```
public class App {
    public static void main(String[] args) throws Exception {
        finalClass obj = new finalClass();
        obj.finalMethod();
    }
}

// final keyword means we cant overwrite the value of any fina
l object in future during writing code other wise it gives an
error "The final local variable finalVariable cannot be assign
ed. It must be blank and not using a compound" assignment
final class finalClass{
    final void finalMethod(){
        final int finalVariable=100;
        System.out.println(finalVariable);
    }
}

~
~
~
~
                                        3,10          All
```

Output:


src : zsh — Konsole
┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q16/src]
└─$ java App
1 ⚙
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
100
┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q16/src]
└─$
1 ⚙

Q17. WAP to explain the concept of finalization.

Code:

```
// Java code to show the
// overriding of finalize() method

import java.lang.*;

// Defining a class demo since every java class
// is a subclass of predefined Object class
// Therefore demo is a subclass of Object class
public class App {

protected void finalize() throws Throwable
{
try {

System.out.println("inside demo's finalize()");
}
catch (Throwable e) {

throw e;
}
finally {

System.out.println("Calling finalize method"
+ " of the Object class");

// Calling finalize() of Object class
super.finalize();
}
}

// Driver cde
public static void main(String[] args) throws Throwable
{

// Creating demo's object
App d = new App();

// Calling finalize of demo
d.finalize();
```

```
}
}
```

src : vim — Konsole

```
import java.lang.*;

// Defining a class demo since every java class
// is a subclass of predefined Object class
// Therefore demo is a subclass of Object class
public class App {

        protected void finalize() throws Throwable
        {
                try {

                        System.out.println("inside demo's fina
lize()");
                }
                catch (Throwable e) {

                        throw e;
                }
                finally {

                        System.out.println("Calling finalize m
ethod"
                                                16,1-8        14%
```

```
                        System.out.println("Calling finalize met
hod"
                                                        + " of t
he Object class");

                        // Calling finalize() of Object class
                        super.finalize();
                }
        }

        // Driver code
        public static void main(String[] args) throws Throwable
        {

                // Creating demo's object
                App d = new App();

                // Calling finalize of demo
                d.finalize();
        }
}
~
                                                33,1-8        Bot
```

Output:

```
┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q17/src]
└─$ java App                                                    4 ⚙
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswin
g.aatext=true
inside demo's finalize()
Calling finalize method of the Object class

┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q17/src]
└─$                                                             4 ⚙
```

Q18. WAP to demonstrate the use of abstract methods and abstract classes.

Code:

```
// A java program to demonstrate
// use of abstract keyword.

// abstract class
abstract class A {
   // abstract method
   // it has no body
   abstract void m1();

   // concrete methods are still
   // allowed in abstract classes
   void m2() {
      System.out.println("This is "
            + "a concrete method.");
   }
}

// concrete class B
class B extends A {
   // class B must override m1() method
   // otherwise, compile-time
   // exception will be thrown
   void m1() {
      System.out.println("B's "
            + "implementation of m1.");
   }
}

// Driver class
public class App {
   public static void main(String args[]) {
      B b = new B();
      b.m1();
      b.m2();
   }
}
```

src : vim — Konsole

```
// abstract class
abstract class A {
    // abstract method
    // it has no body
    abstract void m1();

    // concrete methods are still
    // allowed in abstract classes
    void m2() {
        System.out.println("This is "
                + "a concrete method.");
    }
}

// concrete class B
class B extends A {
    // class B must override m1() method
    // otherwise, compile-time
    // exception will be thrown
    void m1() {
        System.out.println("B's "
                + "implementation of m1.");
    }
                                                18,1          23%
```

src : vim — Konsole

```
    // exception will be thrown
    void m1() {
        System.out.println("B's "
                + "implementation of m1.");
    }
}

// Driver class
public class App {
    public static void main(String args[]) {
        B b = new B();
        b.m1();
        b.m2();
    }
}
~
~
~
~
~
~
~
~
                                        27,1          Bot
```

Output:

```
┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q18/src]
└─$ java App                                                    5 ⚙
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswin
g.aatext=true
B's implementation of m1.
This is a concrete method.

┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q18/src]
└─$                                                             5 ⚙
```

Q19. WAP to demonstrate the use of access modifiers (Private, protected, public and default)

Code:

```
public class Public {
   public String name;
   void Name(){
      System.out.println(name);
   }
}
public class Private {
   private int age;
   public int getAge(){
      return age;
   }
   public void setAge(int Age){
      this.age = Age;
   }
}
public class Protected extends ProtectedClass{

}
class ProtectedClass{
   protected void display() {
      System.out.println("hello");
   }
}
public class App {
   public static void main(String[] args) throws Exception {
      // public access modifier
      Public publicObj = new Public();
      publicObj.name = "Sahil khan";
      publicObj.Name();

      // Private access modifier
      Private privateObj = new Private();
      privateObj.setAge(10);
      System.out.println(privateObj.getAge());
```

```
        // protected acccess modifier
        Protected obj = new Protected();
        obj.display();
    }
}
```

src : vim — Konsole

```
public class Private {
    private int age;
    public int getAge(){
        return age;
    }
    public void setAge(int Age){
        this.age = Age;
    }
}
~
~
~
~
~
~
~
~
~
~
~
~
~
"Private.java" 9L, 160B                    1,1           All
```

src : vim — Konsole

```
public class Protected extends ProtectedClass{

}
class ProtectedClass{
    protected void display() {
        System.out.println("hello");
    }
}
~
~
~
~
~
~
~
~
~
~
~
~
~
~
"Protected.java" [noeol] 8L, 151B            1,1           All
```

src : vim — Konsole

```
public class Public {
    public String name;
    void Name(){
        System.out.println(name);
    }
}
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
"Public.java" 6L, 105B                    1,1           All
```

src : vim — Konsole

```
public class App {
    public static void main(String[] args) throws Exception {
        // public access modifier
        Public publicObj = new Public();
        publicObj.name = "Sahil khan";
        publicObj.Name();

        // Private access modifier
        Private privateObj = new Private();
        privateObj.setAge(10);
        System.out.println(privateObj.getAge());

        // protected acccess modifier
        Protected obj = new Protected();
        obj.display();
    }
}
~
~
~
~
~
~
"App.java" [readonly] 17L, 492B                1,1           All
```

Output:

```
src : zsh — Konsole

┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q19/src]
└─$ java App                                                    10 ⚙
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswin
g.aatext=true
Sahil khan
10
hello

┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q19/src]
└─$                                                             10 ⚙
```

Q20. WA menu driven program to add, subtract and multiply two matrices.

Code:

```
public class AddMatrix {
   AddMatrix(){
      int a[][]={
         {1,3,4},
         {2,3,4},
         {3,4,5}
      };
      int b[][]={
         {1,5,8},
         {10,5,6},
         {9,7,1}
      };
      int c[][] = new int[3][3];
      for(int i=0;i<3;i++){
         for(int j=0;j<3;j++){
            c[i][j] = a[i][j]+b[i][j];
            System.out.print(c[i][j]+"    ");
         }
         System.out.println();
      }
   }
}
public class SubMatrix {
   SubMatrix(){
      int a[][]={
         {1,3,4},
         {2,3,4},
         {3,4,5}
      };
      int b[][]={
         {1,5,8},
         {10,5,6},
         {9,7,1}
      };
      int c[][] = new int[3][3];
```

```
        for(int i=0;i<3;i++){
            for(int j=0;j<3;j++){
                c[i][j] = a[i][j]-b[i][j];
                System.out.print(c[i][j]+"    ");
            }
            System.out.println();
        }
    }
}
public class MulMatrix {
    MulMatrix() {
        int a[][] = { { 1, 1, 1 }, { 2, 2, 2 }, { 3, 3, 3 } };
        int b[][] = { { 1, 1, 1 }, { 2, 2, 2 }, { 3, 3, 3 } };

        // creating another matrix to store the multiplication of two matrices
        int c[][] = new int[3][3]; // 3 rows and 3 columns

        // multiplying and printing multiplication of 2 matrices
        for (int i = 0; i < 3; i++) {
            for (int j = 0; j < 3; j++) {
                c[i][j] = 0;
                for (int k = 0; k < 3; k++) {
                    c[i][j] += a[i][k] * b[k][j];
                } // end of k loop
                System.out.print(c[i][j] + " "); // printing matrix element
            } // end of j loop
            System.out.println();// new line
        }
    }
}
public class App {
    public static void main(String[] args) throws Exception {
        new AddMatrix();
        System.out.println();
        new SubMatrix();
        System.out.println();
        new MulMatrix();
    }
}
```

src : vim — Konsole

```
public class AddMatrix {
    AddMatrix(){
        int a[][]={
            {1,3,4},
            {2,3,4},
            {3,4,5}
        };
        int b[][]={
            {1,5,8},
            {10,5,6},
            {9,7,1}
        };
        int c[][] = new int[3][3];
        for(int i=0;i<3;i++){
            for(int j=0;j<3;j++){
                c[i][j] = a[i][j]+b[i][j];
                System.out.print(c[i][j]+"    ");
            }
            System.out.println();
        }
    }
                                        1,1           Top
```

src : vim — Konsole

```
            {1,3,4},
            {2,3,4},
            {3,4,5}
        };
        int b[][]={
            {1,5,8},
            {10,5,6},
            {9,7,1}
        };
        int c[][] = new int[3][3];
        for(int i=0;i<3;i++){
            for(int j=0;j<3;j++){
                c[i][j] = a[i][j]+b[i][j];
                System.out.print(c[i][j]+"    ");
            }
            System.out.println();
        }
    }
}
~
~
                                        15,1          Bot
```

src : vim — Konsole

```
public class SubMatrix {
    SubMatrix(){
        int a[][]={
            {1,3,4},
            {2,3,4},
            {3,4,5}
        };
        int b[][]={
            {1,5,8},
            {10,5,6},
            {9,7,1}
        };
        int c[][] = new int[3][3];
        for(int i=0;i<3;i++){
            for(int j=0;j<3;j++){
                c[i][j] = a[i][j]-b[i][j];
                System.out.print(c[i][j]+"    ");
            }
            System.out.println();
        }
    }
}
                                                9,1           All
```

src : vim — Konsole

```java
public class MulMatrix {
    MulMatrix() {
        int a[][] = { { 1, 1, 1 }, { 2, 2, 2 }, { 3, 3, 3 } };
        int b[][] = { { 1, 1, 1 }, { 2, 2, 2 }, { 3, 3, 3 } };

        // creating another matrix to store the multiplication of two matrices
        int c[][] = new int[3][3]; // 3 rows and 3 columns

        // multiplying and printing multiplication of 2 matrices
        for (int i = 0; i < 3; i++) {
            for (int j = 0; j < 3; j++) {
                c[i][j] = 0;
                for (int k = 0; k < 3; k++) {
                    c[i][j] += a[i][k] * b[k][j];
                } // end of k loop
                System.out.print(c[i][j] + " "); // printing matrix element
            } // end of j loop
            System.out.println();// new line
        }
    }
}
~
~
                                                    15,1          All
```

```
src : vim — Konsole
public class App {
    public static void main(String[] args) throws Exception {
        new AddMatrix();
        System.out.println();
        new SubMatrix();
        System.out.println();
        new MulMatrix();
    }
}
~
~
~
~
~
~
~
~
~
~
~
~
~
"App.java" 9L, 224B                                    1,1           All
```

Output:

src : vim — Konsole

```
public class App {
    public static void main(String[] args) throws Exception {
        new AddMatrix();
        System.out.println();
        new SubMatrix();
        System.out.println();
        new MulMatrix();
    }
}
~
~
~
~
~
~
~
~
~
~
~
~
~
"App.java" 9L, 224B                                    1,1           All
```

Q21. WA menu driven program for selection sort, merge sort and quick sort.

Code:

```
public class SelectionSort {
void sort(int arr[]){
int n = arr.length;
// One by one move boundary of unsorted subarray
for (int i = 0; i < n-1; i++){
// Find the minimum element in unsorted array
int min_idx = i;
for (int j = i+1; j < n; j++)
if (arr[j] < arr[min_idx])
min_idx = j;

// Swap the found minimum element with the first
// element
int temp = arr[min_idx];
arr[min_idx] = arr[i];
arr[i] = temp;
}
}
// Prints the array
void printArray(int arr[])
{
int n = arr.length;
for (int i=0; i<n; ++i)
System.out.print(arr[i]+" ");
System.out.println();
}
}
public class QuickSort {
/* This function takes last element as pivot,
places the pivot element at its correct
position in sorted array, and places all
smaller (smaller than pivot) to left of
pivot and all greater elements to right
of pivot */
int partition(int arr[], int low, int high){
```

```
int pivot = arr[high];
int i = (low-1); // index of smaller element
for (int j=low; j<high; j++){
// If current element is smaller than or
// equal to pivot
if (arr[j] <= pivot){
i++;
// swap arr[i] and arr[j]
int temp = arr[i];
arr[i] = arr[j];
arr[j] = temp;
}
}
// swap arr[i+1] and arr[high] (or pivot)
int temp = arr[i+1];
arr[i+1] = arr[high];
arr[high] = temp;
return i+1;
}

/* The main function that implements QuickSort()
arr[] --> Array to be sorted,
low --> Starting index,
high --> Ending index */
void sort(int arr[], int low, int high){
if (low < high){
/* pi is partitioning index, arr[pi] is
now at right place */
int pi = partition(arr, low, high);
// Recursively sort elements before
// partition and after partition
sort(arr, low, pi-1);
sort(arr, pi+1, high);
}
}
/* A utility function to print array of size n */
static void printArray(int arr[]){
int n = arr.length;
```

```
for (int i=0; i<n; ++i)
System.out.print(arr[i]+" ");
System.out.println();
}

}
public class MergeSort {
// Merges two subarrays of arr[].
// First subarray is arr[l..m]
// Second subarray is arr[m+1..r]
void merge(int arr[], int l, int m, int r){
// Find sizes of two subarrays to be merged
int n1 = m - l + 1, n2 = r - m;

/* Create temp arrays */
int L[] = new int [n1], R[] = new int [n2];

/*Copy data to temp arrays*/
for (int i=0; i<n1; ++i)
L[i] = arr[l + i];
for (int j=0; j<n2; ++j)
R[j] = arr[m + 1+ j];
/* Merge the temp arrays */

// Initial indexes of first and second subarrays
int i = 0, j = 0;

// Initial index of merged subarray array
int k = l;
while (i < n1 && j < n2){
if (L[i] <= R[j]){
arr[k] = L[i];i++;
}
else{
arr[k] = R[j];j++;
}k++;
}

/* Copy remaining elements of L[] if any */
while (i < n1){
```

```
arr[k] = L[i];i++;k++;
}

/* Copy remaining elements of R[] if any */
while (j < n2){
arr[k] = R[j];j++;k++;
}
}

// Main function that sorts arr[l..r] using
// merge()
void sort(int arr[], int l, int r){
if (l < r)
{
// Find the middle point
int m = (l+r)/2;

// Sort first and second halves
sort(arr, l, m);sort(arr , m+1, r);

// Merge the sorted halves
merge(arr, l, m, r);
}
}

/* A utility function to print array of size n */
static void printArray(int arr[]){
int n = arr.length;
for (int i=0; i<n; ++i)
System.out.print(arr[i] + " ");
System.out.println();
}
}
public class App {
   public static void main(String[] args) throws Exception {
      SelectionSort ob = new SelectionSort();
int arr[] = {64,25,12,22,11};
ob.sort(arr);
System.out.println("Sorted array");
ob.printArray(arr);
```

```
int n = arr.length;
QuickSort q = new QuickSort();
q.sort(arr, 0, n-1);
System.out.println("sorted array");
QuickSort.printArray(arr);


      int array[] = {12, 11, 13, 5, 6, 7};
System.out.println("Given Array");
MergeSort.printArray(array);
MergeSort ms = new MergeSort();
ms.sort(array, 0, array.length-1);
System.out.println("\nSorted array");
MergeSort.printArray(arr);
   }
}
```

src : vim — Konsole

```
public class MergeSort {
        // Merges two subarrays of arr[].
        // First subarray is arr[l..m]
        // Second subarray is arr[m+1..r]
        void merge(int arr[], int l, int m, int r){
                // Find sizes of two subarrays to be merged
                int n1 = m - l + 1, n2 = r - m;

                /* Create temp arrays */
                int L[] = new int [n1], R[] = new int [n2];

                /*Copy data to temp arrays*/
                for (int i=0; i<n1; ++i)
                        L[i] = arr[l + i];
                for (int j=0; j<n2; ++j)
                        R[j] = arr[m + 1+ j];
                /* Merge the temp arrays */

                // Initial indexes of first and second subarrays
                int i = 0, j = 0;

                // Initial index of merged subarray array
                int k = l;
                                                        18,0-1        Top
```

```
src : vim — Konsole
        // Initial index of merged subarray array
        int k = l;
        while (i < n1 && j < n2){
                if (L[i] ≤ R[j]){
                        arr[k] = L[i];i++;
                }
                else{
                        arr[k] = R[j];j++;
                }k++;
        }

        /* Copy remaining elements of L[] if any */
        while (i < n1){
                arr[k] = L[i];i++;k++;
        }

        /* Copy remaining elements of R[] if any */
        while (j < n2){
                arr[k] = R[j];j++;k++;
        }
}

// Main function that sorts arr[l..r] using
                                                27,1-8        47%
```

src : vim — Konsole

```
// Main function that sorts arr[l..r] using
// merge()
void sort(int arr[], int l, int r){
        if (l < r)
        {
                // Find the middle point
                int m = (l+r)/2;

                // Sort first and second halves
                sort(arr, l, m);sort(arr , m+1, r);

                // Merge the sorted halves
                merge(arr, l, m, r);
        }
}

/* A utility function to print array of size n */
static void printArray(int arr[]){
        int n = arr.length;
        for (int i=0; i<n; ++i)
                System.out.print(arr[i] + " ");
        System.out.println();
                                                        48,1-8        95%
```

src : vim — Konsole

```
        }

        /* A utility function to print array of size n */
        static void printArray(int arr[]){
                int n = arr.length;
                for (int i=0; i<n; ++i)
                        System.out.print(arr[i] + " ");
                System.out.println();
        }
}
~
~
~
~
~
~
~
~
~
~
~
~
~
                                                        63,1-8        Bot
```

src : vim — Konsole

```
public class SelectionSort {
        void sort(int arr[]){
                int n = arr.length;
                // One by one move boundary of unsorted subarray
                for (int i = 0; i < n-1; i++){
                        // Find the minimum element in unsorted array
                        int min_idx = i;
                        for (int j = i+1; j < n; j++)
                                if (arr[j] < arr[min_idx])
                                        min_idx = j;

                        // Swap the found minimum element with the first
                        // element
                        int temp = arr[min_idx];
                        arr[min_idx] = arr[i];
                        arr[i] = temp;
                }
        }
        // Prints the array
        void printArray(int arr[])
        {
                int n = arr.length;
                for (int i=0; i<n; ++i)
                                                        18,1-8        Top
```

```
src : vim — Konsole
        // Prints the array
        void printArray(int arr[])
        {
                int n = arr.length;
                for (int i=0; i<n; ++i)
                        System.out.print(arr[i]+" ");
                System.out.println();
        }
}
~
~
~
~
~
~
~
~
~
~
~
~
~
                                                24,1-8        Bot
```

src : vim — Konsole

```
public class QuickSort {
        /* This function takes last element as pivot,
        places the pivot element at its correct
        position in sorted array, and places all
        smaller (smaller than pivot) to left of
        pivot and all greater elements to right
        of pivot */
        int partition(int arr[], int low, int high){
                int pivot = arr[high];
                int i = (low-1); // index of smaller element
                for (int j=low; j<high; j++){
                        // If current element is smaller than or
                        // equal to pivot
                        if (arr[j] ≤ pivot){
                                i++;
                                // swap arr[i] and arr[j]
                                int temp = arr[i];
                                arr[i] = arr[j];
                                arr[j] = temp;
                        }
                }
                // swap arr[i+1] and arr[high] (or pivot)
                int temp = arr[i+1];
"QuickSort.java" 53L, 1363B                                1,1           Top
```

```
src : vim — Konsole
        // swap arr[i+1] and arr[high] (or pivot)
        int temp = arr[i+1];
        arr[i+1] = arr[high];
        arr[high] = temp;
        return i+1;
}


/* The main function that implements QuickSort()
arr[] ⟶ Array to be sorted,
low ⟶ Starting index,
high ⟶ Ending index */
void sort(int arr[], int low, int high){
        if (low < high){
                /* pi is partitioning index, arr[pi] is
                now at right place */
                int pi = partition(arr, low, high);
                // Recursively sort elements before
                // partition and after partition
                sort(arr, low, pi-1);
                sort(arr, pi+1, high);
        }
}
                                                        30,1-8        70%
```

```
                        // partition and after partition
                        sort(arr, low, pi-1);
                        sort(arr, pi+1, high);
                }
        }
        /* A utility function to print array of size n */
        static void printArray(int arr[]){
                int n = arr.length;
                for (int i=0; i<n; ++i)
                        System.out.print(arr[i]+" ");
                System.out.println();
        }

}
~
~
~
~
~
~
~
~
~
                                                        45,1-8        Bot
```

src : vim — Konsole

```
public class App {
    public static void main(String[] args) throws Exception {
        SelectionSort ob = new SelectionSort();
                int arr[] = {64,25,12,22,11};
                ob.sort(arr);
                System.out.println("Sorted array");
                ob.printArray(arr);


                int n = arr.length;
                QuickSort q = new QuickSort();
                q.sort(arr, 0, n-1);
                System.out.println("sorted array");
                QuickSort.printArray(arr);


        int array[] = {12, 11, 13, 5, 6, 7};
                System.out.println("Given Array");
                MergeSort.printArray(array);
                MergeSort ms = new MergeSort();
                ms.sort(array, 0, array.length-1);
                System.out.println("\nSorted array");
                MergeSort.printArray(arr);
                                                        18,1-8        Top
```

```
src : vim — Konsole
                int n = arr.length;
                QuickSort q = new QuickSort();
                q.sort(arr, 0, n-1);
                System.out.println("sorted array");
                QuickSort.printArray(arr);


        int array[] = {12, 11, 13, 5, 6, 7};
                System.out.println("Given Array");
                MergeSort.printArray(array);
                MergeSort ms = new MergeSort();
                ms.sort(array, 0, array.length-1);
                System.out.println("\nSorted array");
                MergeSort.printArray(arr);
    }
}
~
~
~
~
~
~
~
                                                18,1-8        Bot
```

Output:

```
┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q21/src]
└─$ java App                                                          18 ⚙
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Sorted array
11 12 22 25 64
sorted array
11 12 22 25 64
Given Array
12 11 13 5 6 7

Sorted array
11 12 22 25 64

┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q21/src]
└─$                                                                   18 ⚙
```

Q22. WAP in which you have to make a table to store name of students, age, course, roll no and this table can be ordered in following ways:

1: Alphabetical order (A to Z or Z to A) of name

2: smallest to greatest or greatest to smallest order of age

3: Ascending or descending order of roll no

4: Bachelor to master or master to bachelor order of course

Code:

```
import java.util.*;

public class App {

  public static void main(String[] args) throws Exception {
    int n =4;
    String names[]={"Bharti","Ziya","Rufia","Kisa"};
    String temp;
    for (int i=0;i<=n;i++){
      for(int j=i+1;j<n;j++){
        if (names[i].compareTo(names[j])>0){
          //swaping
          temp = names[i];
          names[i]=names[j];
          names[j] = temp;
        }
      }
    }
    // print output array
    System.out.println("The names in alphabetical order are :: ");
    for(int i=0;i<n;i++){
      System.out.println(names[i]);
    }
  }
}
```

src : vim — Konsole

```java
import java.util.*;

public class App {

  public static void main(String[] args) throws Exception {
    int n =4;
    String names[]={"Bharti","Ziya","Rufia","Kisa"};
    String temp;
    for (int i=0;i<n;i++){
      for(int j=i+1;j<n;j++){
        if (names[i].compareTo(names[j])>0){
          //swaping
          temp = names[i];
          names[i]=names[j];
          names[j] = temp;
        }
      }
    }
    // print output array
    System.out.println("The names in alphabetical order are :: ");
    for(int i=0;i<n;i++){
      System.out.println(names[i]);
    }
  }
}
                                                    21,1          All
```

Output:

```
src : zsh — Konsole
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
The names in alphabetical order are ::
Bharti
Kisa
Rufia
Ziya

┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q22/src]
└─$                                                                    43 ⚙
```

Q23. WAP to demonstrate the commonly used "String" and "StringBuffer" class methods.

Code:

```
import java.util.*;

public class App {

  public static void main(String[] args) throws Exception {
    int n =4;
    String names[]={"Bharti","Ziya","Rufia","Kisa"};
    String temp;
    for (int i=0;i<=n;i++){
      for(int j=i+1;j<n;j++){
        if (names[i].compareTo(names[j])>0){
          //swaping
          temp = names[i];
          names[i]=names[j];
          names[j] = temp;
        }
      }
    }
    // print output array
    System.out.println("The names in alphabetical order are :: ");
    for(int i=0;i<n;i++){
      System.out.println(names[i]);
    }
  }
}
```

src : vim — Konsole

```java
public class App {
    public static void main(String[] args) throws Exception {
        StringBuffer sb = new StringBuffer("Hello ");

        // append
        sb.append("Java"); // now original string is changed
        System.out.println(sb);

        // insert
        sb.insert(1, "Java");
        System.out.println(sb);

        // replace
        sb.replace(1, 3, "Java");
        System.out.println(sb);

        // delete
        sb.delete(1, 3);
        System.out.println(sb);

                                                    15,31         Top
```

src : vim — Konsole

```
        // delete
        sb.delete(1, 3);
        System.out.println(sb);

        // reverse
        sb.reverse();
        System.out.println(sb);

        // capacity
        System.out.println(sb.capacity()); //default 16
        sb.append("hello");
        System.out.println(sb.capacity()); //now 16
        sb.append("java is my favourite languge");
        System.out.println(sb.capacity());
        // Now (16*2)+2 = 34 i.e (oldcapacity*2)+2

    }
}
~
                                                    21,18         Bot
```

Output:

```
┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q23/src]
└─$ java App                                                    19 ⚙
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatex
t=true
Hello Java
HJavaello Java
HJavavaello Java
Hvavaello Java
avaJ olleavavH
22
22
47

┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q23/src]
└─$                                                             19 ⚙
```

Q24. WAP to demonstrate Wrapper class methods.

Code:

```
public class App {
    public static void main(String[] args) throws Exception {
        byte b = 10;
        short s = 20;
        int i = 30;
        long l = 40;
        float f = 50.0F;
        double d = 60.0D;
        char c = 'a';
        boolean b2 = true;

        // Autoboxing: Converting primitives into objects
        Byte byteobj = b;
        Short shortobj = s;
        Integer intobj = i;
        Long longobj = l;
        Float floatobj = f;
        Double doubleobj = d;
        Character charobj = c;
        Boolean boolobj = b2;

        // Printing objects
        System.out.println("---Printing object values---");
        System.out.println("Byte object: " + byteobj);
        System.out.println("Short object: " + shortobj);
        System.out.println("Integer object: " + intobj);
        System.out.println("Long object: " + longobj);
        System.out.println("Float object: " + floatobj);
        System.out.println("Double object: " + doubleobj);
        System.out.println("Character object: " + charobj);
        System.out.println("Boolean object: " + boolobj);

        // Unboxing: Converting Objects to Primitives
        byte bytevalue = byteobj;
        short shortvalue = shortobj;
        int intvalue = intobj;
```

```
        long longvalue = longobj;
        float floatvalue = floatobj;
        double doublevalue = doubleobj;
        char charvalue = charobj;
        boolean boolvalue = boolobj;

        // Printing primitives
        System.out.println("---Printing primitive values---");
        System.out.println("byte value: " + bytevalue);
        System.out.println("short value: " + shortvalue);
        System.out.println("int value: " + intvalue);
        System.out.println("long value: " + longvalue);
        System.out.println("float value: " + floatvalue);
        System.out.println("double value: " + doublevalue);
        System.out.println("char value: " + charvalue);
        System.out.println("boolean value: " + boolvalue);
    }
}
```

src : vim — Konsole

```
public class App {
    public static void main(String[] args) throws Exception {
        byte b = 10;
        short s = 20;
        int i = 30;
        long l = 40;
        float f = 50.0F;
        double d = 60.0D;
        char c = 'a';
        boolean b2 = true;

        // Autoboxing: Converting primitives into objects
        Byte byteobj = b;
        Short shortobj = s;
        Integer intobj = i;
        Long longobj = l;
        Float floatobj = f;
        Double doubleobj = d;
        Character charobj = c;
        Boolean boolobj = b2;
                                                    15,1          Top
```

src : vim — Konsole

```
        Character charobj = c;
        Boolean boolobj = b2;

        // Printing objects
        System.out.println("——Printing object values——");
        System.out.println("Byte object: " + byteobj);
        System.out.println("Short object: " + shortobj);
        System.out.println("Integer object: " + intobj);
        System.out.println("Long object: " + longobj);
        System.out.println("Float object: " + floatobj);
        System.out.println("Double object: " + doubleobj);
        System.out.println("Character object: " + charobj);
        System.out.println("Boolean object: " + boolobj);

        // Unboxing: Converting Objects to Primitives
        byte bytevalue = byteobj;
        short shortvalue = shortobj;
        int intvalue = intobj;
        long longvalue = longobj;
        float floatvalue = floatobj;
                                                    24,1          52%
```

```
        // Unboxing: Converting Objects to Primitives
        byte bytevalue = byteobj;
        short shortvalue = shortobj;
        int intvalue = intobj;
        long longvalue = longobj;
        float floatvalue = floatobj;
        double doublevalue = doubleobj;
        char charvalue = charobj;
        boolean boolvalue = boolobj;

        // Printing primitives
        System.out.println("——Printing primitive values——");
        System.out.println("byte value: " + bytevalue);
        System.out.println("short value: " + shortvalue);
        System.out.println("int value: " + intvalue);
        System.out.println("long value: " + longvalue);
        System.out.println("float value: " + floatvalue);
        System.out.println("double value: " + doublevalue);
        System.out.println("char value: " + charvalue);
        System.out.println("boolean value: " + boolvalue);
    }
}
                                                    39,1          Bot
```

Output:

```
┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q24/src]
└─$ java App                                                    20 ⚙
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=t
rue
──Printing object values──
Byte object: 10
Short object: 20
Integer object: 30
Long object: 40
Float object: 50.0
Double object: 60.0
Character object: a
Boolean object: true
──Printing primitive values──
byte value: 10
short value: 20
int value: 30
long value: 40
float value: 50.0
double value: 60.0
char value: a
boolean value: true
```

Q25. WAP to explain interfaces in java.

Code:

```
public class App {
   public static void main(String[] args) throws Exception {
      Hello h1 = new Hello();
      h1.print();
   }
}
/**
* Printable
*/
interface Printable {
   void print();
}
class Hello implements Printable{
   @Override
   public void print() {
      System.out.println("hello world!");
   }
}
```

```
src : vim — Konsole
public class App {
    public static void main(String[] args) throws Exception {
        Hello h1 = new Hello();
        h1.print();
    }
}
/**
 * Printable
 */
interface Printable {
    void print();
}
class Hello implements Printable{
    @Override
    public void print() {
        System.out.println("hello world!");
    }
}
~
~
~
~
                                                    12,1          All
```

Output:

```
src : zsh — Konsole

┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q25/src]
└─$ java App                                                    21 ⚙
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=t
rue
hello world!

┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q25/src]
└─$                                                             21 ⚙
```

Q26. WAP to demonstrate multiple inheritances in java.

Code:

```
import java.io.*;
public class App extends Parent1,Parent2 {
   public static void main(String[] args) throws Exception {
      System.out.println("Hello, World!");
   }
}

//  Class 1
// A Grand parent class in diamond
class GrandParent {

  void fun() {

    // Print statement to be executed when this method is called
    System.out.println("Grandparent");
  }
}

// Class 2
// First Parent class
class Parent1 extends GrandParent {
  void fun() {

    // Print statement to be executed when this method is called
    System.out.println("Parent1");
  }
}

// Class 3
// Second Parent Class
class Parent2 extends GrandParent {
  void fun() {

    // Print statement to be executed when this method is called
    System.out.println("Parent2");
  }
}
```

src : vim — Konsole

```
import java.io.*;
public class App extends Parent1,Parent2 {
    public static void main(String[] args) throws Exception {
        System.out.println("Hello, World!");
    }
}

//  Class 1
// A Grand parent class in diamond
class GrandParent {

  void fun() {

    // Print statement to be executed when this method is called
    System.out.println("Grandparent");
  }
}

// Class 2
// First Parent class
class Parent1 extends GrandParent {
  void fun() {
                                                    17,1          Top
```

```
  }
}

// Class 2
// First Parent class
class Parent1 extends GrandParent {
  void fun() {

    // Print statement to be executed when this method is called
    System.out.println("Parent1");
  }
}

// Class 3
// Second Parent Class
class Parent2 extends GrandParent {
  void fun() {

    // Print statement to be executed when this method is called
    System.out.println("Parent2");
  }
}
                                                    21,1          Bot
```

Output:

```
┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q26/src]
└─$ java App                                                    23 ⚙
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=t
rue
Error: Could not find or load main class App
Caused by: java.lang.ClassNotFoundException: App

┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q26/src]
└─$                                                             1 × 23 ⚙
```

Note: multiply inheritance not supported in java latest version

Q27. WAP to demonstrate package in java.

and

Q28. WAP to explain static import in java with package.

Code:

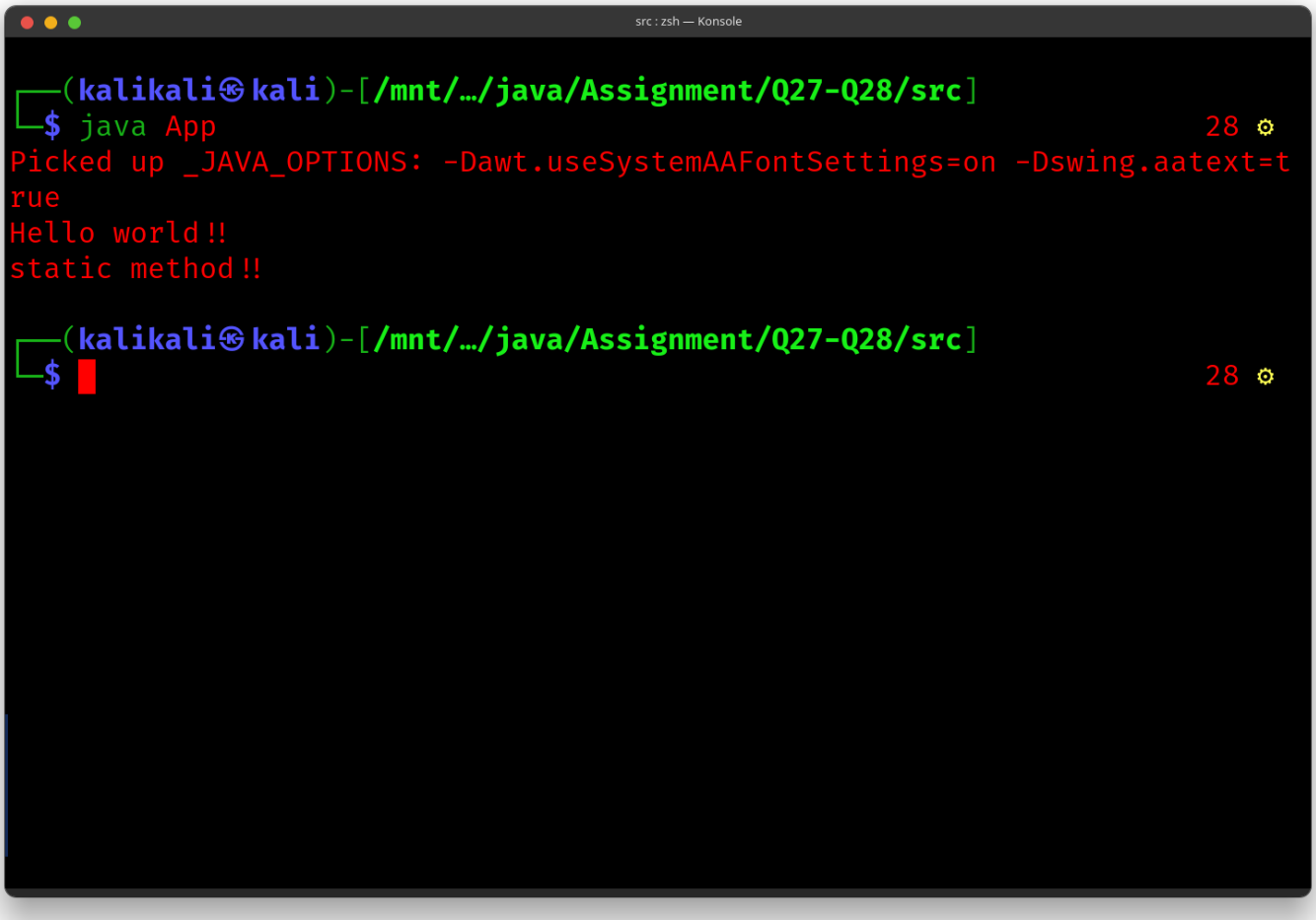
```
import hello.Hello;

public class App {
    public static void main(String[] args) throws Exception {
        new Hello();
        Hello.print();
    }
}
package hello;

public class Hello {
    public  Hello(){
        System.out.println("Hello world!!");
    }
    public static void print(){
        System.out.println("static method!!");
    }
}
```

hello : vim — Konsole

```
package hello;

public class Hello {
    public  Hello(){
        System.out.println("Hello world!!");
    }
    public static void print(){
        System.out.println("static method!!");
    }
}
~
~
~
~
~
~
~
~
~
~
~
~
"Hello.java" 10L, 196B                                    1,1           All
```

```
import hello.Hello;

public class App {
    public static void main(String[] args) throws Exception {
        new Hello();
        Hello.print();
    }
}
~
~
~
~
~
~
~
~
~
~
~
~
~
"App.java" [readonly] 8L, 154B                    1,1           All
```

Output:


src : zsh — Konsole
┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q27-Q28/src]
└─$ java App
28 ⚙
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Hello world!!
static method!!
┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q27-Q28/src]
└─$
28 ⚙

Q29. WAP to demonstrate threading in java using Thread class.

and

Q30. WAP to demonstrate threading in java using Runnable interface.

Code:

```
public class App extends Thread {
    public static void main(String[] args) throws Exception {
        App thread = new App();
        thread.start();
        System.out.println("This code is outside of the thread");




        Runable run1 = new Runable();
        Thread t1 = new Thread(run1);
        t1.start();
        System.out.println("hi");
    }
    public void run(){
        System.out.println("this code is running in  a thread");
    }
}
class Runable implements Runnable{
    public void run(){
        System.out.println("thread has ended");
    }
}
```

src : vim — Konsole

```
public class App extends Thread {
    public static void main(String[] args) throws Exception {
        App thread = new App();
        thread.start();
        System.out.println("This code is outside of the thread");




        Runable run1 = new Runable();
        Thread t1 = new Thread(run1);
        t1.start();
        System.out.println("hi");
    }
    public void run(){
        System.out.println("this code is running in  a thread");
    }


}
class Runable implements Runnable{
    public void run(){
                                                        17,1          Top
```

```
        System.out.println("hi");
    }
    public void run(){
        System.out.println("this code is running in  a thread");
    }


}
class Runable implements Runnable{
    public void run(){
        System.out.println("thread has ended");
    }
}
~
~
~
~
~
~
~
~
~
                                                    21,1          Bot
```

Output:

```
┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q29-Q30/src]
└─$ java App                                                    29 ⚙
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=t
rue
This code is outside of the thread
this code is running in  a thread
hi
thread has ended

┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q29-Q30/src]
└─$                                                             29 ⚙
```

Q31. WAP to explain try and catch statements in java.

and

Q32. WAP to demonstrate exception handling using multiple catch statements and finally block.

Code:

```
public class App {

  public static void main(String[] args) throws Exception {
    try {
      int a[] = new int[5];
      a[5] = 30 / 0;
    } catch (ArithmeticException e) {
      System.out.println("Arithmetic Exception occurs");
    } catch (ArrayIndexOutOfBoundsException e) {
      System.out.println("ArrayIndexOutOfBounds Exception occurs");
    } catch (Exception e) {
      System.out.println("Parent Exception occurs");
    }
    finally{
        System.out.println("rest of the code!!");
    }
  }
}
```

```
public class App {

  public static void main(String[] args) throws Exception {
    try {
      int a[] = new int[5];
      a[5] = 30 / 0;
    } catch (ArithmeticException e) {
      System.out.println("Arithmetic Exception occurs");
    } catch (ArrayIndexOutOfBoundsException e) {
      System.out.println("ArrayIndexOutOfBounds Exception occurs");
    } catch (Exception e) {
      System.out.println("Parent Exception occurs");
    }
    finally{
        System.out.println("rest of the code!!");
    }
  }
}
~
~
~
~
                                                    12,1          All
```

Output:

```
┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q31-Q32/src]
└─$ java App                                                    30 ⚙
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=t
rue
Arithmetic Exception occurs
rest of the code!!

┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q31-Q32/src]
└─$                                                             30 ⚙
```

Q33. WAP to demonstrate the use of “throw” and “throws” keyword in java.

Code:

```
public class App {

  static void fun() {
    try {
      throw new NullPointerException("demo");
    } catch (NullPointerException e) {
      System.out.println("Caught inside fun().");
      throw e; // rethrowing the exception
    }
  }

  public static void main(String[] args) throws Exception {
    try {
      fun();
    } catch (NullPointerException e) {
      System.out.println("Caught in main.");
    }
  }
}
```

```
public class App {

  static void fun() {
    try {
      throw new NullPointerException("demo");
    } catch (NullPointerException e) {
      System.out.println("Caught inside fun().");
      throw e; // rethrowing the exception
    }
  }

  public static void main(String[] args) throws Exception {
    try {
      fun();
    } catch (NullPointerException e) {
      System.out.println("Caught in main.");
    }
  }
}
~
~
~
                                                    12,1          All
```

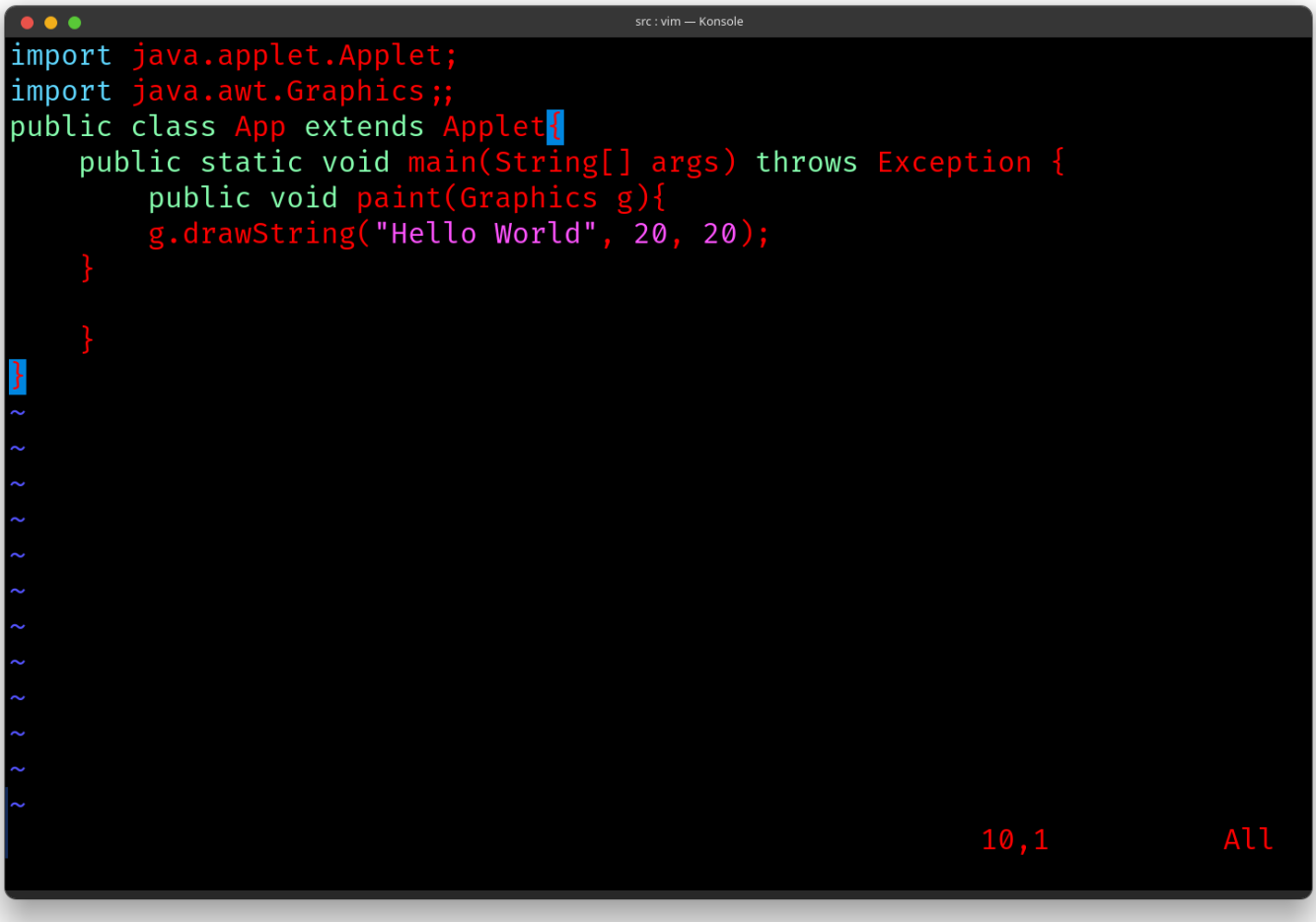
Output:

```
┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q33/src]
└─$ java App                                                    31 ⚙
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=t
rue
Caught inside fun().
Caught in main.

┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q33/src]
└─$                                                             31 ⚙
```

Q34. WAP to demonstrate Applet with all the states used in it.

Code:

```
import java.applet.Applet;
import java.awt.Graphics;;
public class App extends Applet{
   public static void main(String[] args) throws Exception {
      public void paint(Graphics g){
      g.drawString("Hello World", 20, 20);
   }

   }
}
```



Output:

```
┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q34/src]
└─$ ls                                                          31 ⚙
App.java

┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q34/src]
└─$ vim App.java                                                31 ⚙

zsh: suspended  vim App.java

┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q34/src]
└─$ javac App.java                                        148 × 32 ⚙
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=t
rue
App.java:5: error: illegal start of expression
        public void paint(Graphics g){
        ^
App.java:10: error: class, interface, enum, or record expected
}
^
2 errors

┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q34/src]
└─$                                                          1 × 32 ⚙
```

Note: Applet is not supported in latest version of java

Q35. WAP to make graphic calculator.

Code:

```
import javax.swing.*; import java.awt.*; import java.awt.event.*;

class MyFrame extends JFrame implements ActionListener { private Container c; private
JTextField t1;
private JButton b1, b2, b3, b4, b5, b6, b7, b8, b9, b0, ba, bs, bm, bd, bc, beq; int b=0, f=0,
d=0, s=0;
MyFrame(){setTitle("Simple Calculator"); setSize(620, 280); setLocation(200, 200);
setDefaultCloseOperation(EXIT_ON_CLOSE); c=getContentPane(); c.setLayout(null);
t1=new JTextField(); t1.setBounds(20,20,560,60); t1.setBackground(Color.BLACK);
t1.setForeground(Color.CYAN); t1.setEditable(false); c.add(t1); setVisible(true);
b1=new JButton("1"); b1.setBounds(20,110,70,30); c.add(b1);
b1.setBackground(Color.GRAY); b1.setForeground(Color.WHITE);
b2=new JButton("2"); b2.setBounds(100,110,70,30); c.add(b2);
b2.setBackground(Color.GRAY); b2.setForeground(Color.WHITE);
b3=new JButton("3"); b3.setBounds(180,110,70,30); c.add(b3);
b3.setBackground(Color.GRAY); b3.setForeground(Color.WHITE);
b4=new JButton("4"); b4.setBounds(260,110,70,30); c.add(b4);
b4.setBackground(Color.GRAY); b4.setForeground(Color.WHITE);
b5=new JButton("5"); b5.setBounds(20,150,70,30); c.add(b5);
b5.setBackground(Color.GRAY); b5.setForeground(Color.WHITE);
b6=new JButton("6"); b6.setBounds(100,150,70,30); c.add(b6);
b6.setBackground(Color.GRAY); b6.setForeground(Color.WHITE);
b7=new JButton("7"); b7.setBounds(180,150,70,30); c.add(b7);
b7.setBackground(Color.GRAY); b7.setForeground(Color.WHITE);
b8=new JButton("8"); b8.setBounds(260,150,70,30); c.add(b8);
b8.setBackground(Color.GRAY); b8.setForeground(Color.WHITE);
b9=new JButton("9"); b9.setBounds(20,190,70,30); c.add(b9);
b9.setBackground(Color.GRAY); b9.setForeground(Color.WHITE);
b0=new JButton("0"); b0.setBounds(100,190,70,30); c.add(b0);
b0.setBackground(Color.GRAY); b0.setForeground(Color.WHITE);
bd=new JButton("/"); bd.setBounds(500,150,70,30); c.add(bd);
bd.setBackground(Color.BLACK); bd.setForeground(Color.WHITE);
ba=new JButton("+"); ba.setBounds(420,110,70,30); c.add(ba);
ba.setBackground(Color.BLACK); ba.setForeground(Color.WHITE);
```

```
bs=new JButton("-"); bs.setBounds(500,110,70,30); c.add(bs);
bs.setBackground(Color.BLACK); bs.setForeground(Color.WHITE);
bm=new JButton("x"); bm.setBounds(420,150,70,30); c.add(bm);
bm.setBackground(Color.BLACK); bm.setForeground(Color.WHITE);
beq=new JButton("="); beq.setBounds(460,190,70,30); c.add(beq);
beq.setBackground(Color.BLACK); beq.setForeground(Color.WHITE);
bc=new JButton("c"); bc.setBounds(260,190,70,30); c.add(bc);
bc.setBackground(Color.BLACK); bc.setForeground(Color.WHITE);
b1.addActionListener(this); b2.addActionListener(this); b3.addActionListener(this);
b4.addActionListener(this);
b5.addActionListener(this); b6.addActionListener(this); b7.addActionListener(this);
b8.addActionListener(this);
b9.addActionListener(this); b0.addActionListener(this); bc.addActionListener(this);
ba.addActionListener(this);
bs.addActionListener(this); bm.addActionListener(this); bd.addActionListener(this);
beq.addActionListener(this);}
public void actionPerformed(ActionEvent e){int a=0; String t;
if(e.getSource()==bc){t1.setText("");}
if(e.getSource()==b1){t=t1.getText(); a=t.length(); if(a==0){t1.setText("1");}
if(a>0 && a<4){t1.setText(t+"1");}}
if(e.getSource()==b2){t=t1.getText(); a=t.length(); if(a==0){t1.setText("2");}
if(a>0 && a<4){t1.setText(t+"2");}}
if(e.getSource()==b3){t=t1.getText(); a=t.length(); if(a==0){t1.setText("3");}
if(a>0 && a<4){t1.setText(t+"3");}}
if(e.getSource()==b4){t=t1.getText(); a=t.length(); if(a==0){t1.setText("4");}
if(a>0 && a<4){t1.setText(t+"4");}}
if(e.getSource()==b5){t=t1.getText(); a=t.length(); if(a==0){t1.setText("5");}
if(a>0 && a<4){t1.setText(t+"5");}}
if(e.getSource()==b6){t=t1.getText(); a=t.length(); if(a==0){t1.setText("6");}
if(a>0 && a<4){t1.setText(t+"6");}}
if(e.getSource()==b7){t=t1.getText(); a=t.length(); if(a==0){t1.setText("7");}
if(a>0 && a<4){t1.setText(t+"7");}}
if(e.getSource()==b8){t=t1.getText(); a=t.length(); if(a==0){t1.setText("8");}
if(a>0 && a<4){t1.setText(t+"8");}}
if(e.getSource()==b9){t=t1.getText(); a=t.length(); if(a==0){t1.setText("9");}
if(a>0 && a<4){t1.setText(t+"9");}}
if(e.getSource()==b0){t=t1.getText(); a=t.length(); if(a==0){t1.setText("0");}
if(a>0 && a<4){t1.setText(t+"0");}}
```

```
if(e.getSource()==ba){t=t1.getText(); a=t.length(); if(a>0){b=Integer.parseInt(t);
t1.setText(""); s=1;}}
if(e.getSource()==bs){t=t1.getText(); a=t.length(); if(a>0){b=Integer.parseInt(t);
t1.setText(""); s=2;}}
if(e.getSource()==bm){t=t1.getText(); a=t.length(); if(a>0){b=Integer.parseInt(t);
t1.setText(""); s=3;}}
if(e.getSource()==bd){t=t1.getText(); a=t.length(); if(a>0){b=Integer.parseInt(t);
t1.setText(""); s=4;}}
if(e.getSource()==beq){t=t1.getText(); a=t.length(); if(a>0){d=Integer.parseInt(t);
if(s==1){f=b+d;} if(s==2){f=b-d;} if(s==3){f=b*d;} if(s==4){f=b/d;} t1.setText(""+f);}}
}}

public class App {

    public static void main(String[] args) {
        MyFrame frame=new MyFrame();
    }
}
```

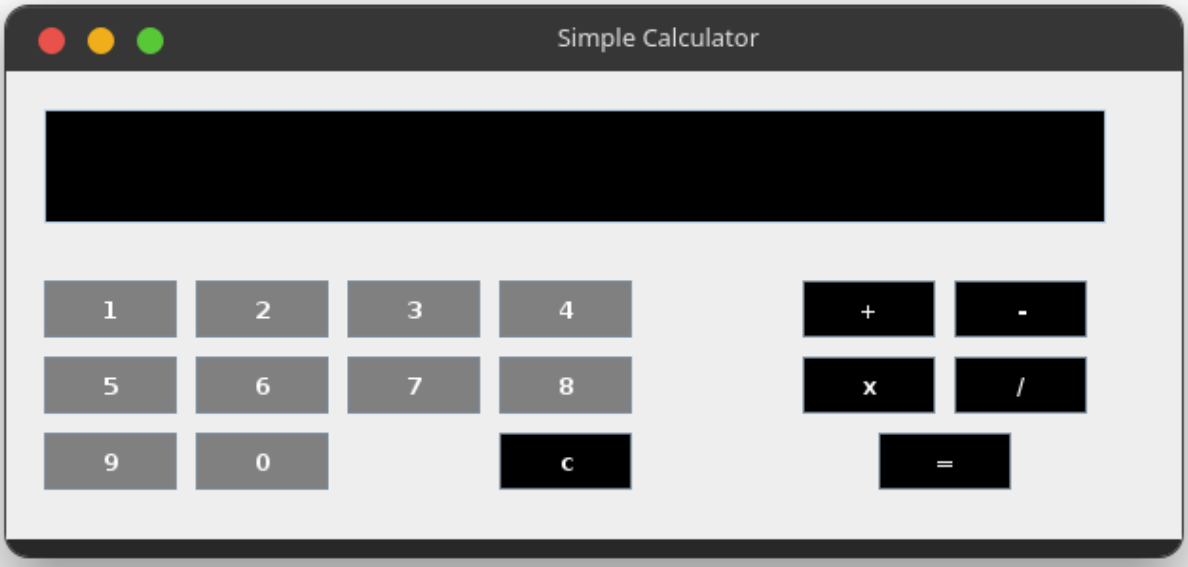
```
import javax.swing.*; import java.awt.*; import java.awt.event.*;

class MyFrame extends JFrame implements ActionListener { private Container c; private JTextField t1;
private JButton b1, b2, b3, b4, b5, b6, b7, b8, b9, b0, ba, bs, bm, bd, bc, beq; int b=0, f=0, d=0, s=0;
MyFrame(){setTitle("Simple Calculator"); setSize(620, 280); setLocation(200, 200);
setDefaultCloseOperation(EXIT_ON_CLOSE); c=getContentPane(); c.setLayout(null);
t1=new JTextField(); t1.setBounds(20,20,560,60); t1.setBackground(Color.BLACK);
t1.setForeground(Color.CYAN); t1.setEditable(false); c.add(t1); setVisible(true);
b1=new JButton("1"); b1.setBounds(20,110,70,30); c.add(b1); b1.setBackground(Color.GRAY); b1.setForegrou
nd(Color.WHITE);
b2=new JButton("2"); b2.setBounds(100,110,70,30); c.add(b2); b2.setBackground(Color.GRAY); b2.setForegro
und(Color.WHITE);
b3=new JButton("3"); b3.setBounds(180,110,70,30); c.add(b3); b3.setBackground(Color.GRAY); b3.setForegro
und(Color.WHITE);
b4=new JButton("4"); b4.setBounds(260,110,70,30); c.add(b4); b4.setBackground(Color.GRAY); b4.setForegro
und(Color.WHITE);
b5=new JButton("5"); b5.setBounds(20,150,70,30); c.add(b5); b5.setBackground(Color.GRAY); b5.setForegrou
nd(Color.WHITE);
b6=new JButton("6"); b6.setBounds(100,150,70,30); c.add(b6); b6.setBackground(Color.GRAY); b6.setForegro
und(Color.WHITE);
b7=new JButton("7"); b7.setBounds(180,150,70,30); c.add(b7); b7.setBackground(Color.GRAY); b7.setForegro
und(Color.WHITE);
b8=new JButton("8"); b8.setBounds(260,150,70,30); c.add(b8); b8.setBackground(Color.GRAY); b8.setForegro
und(Color.WHITE);
b9=new JButton("9"); b9.setBounds(20,190,70,30); c.add(b9); b9.setBackground(Color.GRAY); b9.setForegrou
nd(Color.WHITE);
                                                                                                  13,1          Top
```

```
b8=new JButton("8"); b8.setBounds(260,150,70,30); c.add(b8); b8.setBackground(Color.GRAY); b8.setForegro
und(Color.WHITE);
b9=new JButton("9"); b9.setBounds(20,190,70,30); c.add(b9); b9.setBackground(Color.GRAY); b9.setForegrou
nd(Color.WHITE);
b0=new JButton("0"); b0.setBounds(100,190,70,30); c.add(b0); b0.setBackground(Color.GRAY); b0.setForegro
und(Color.WHITE);
bd=new JButton("/"); bd.setBounds(500,150,70,30); c.add(bd); bd.setBackground(Color.BLACK); bd.setForegr
ound(Color.WHITE);
ba=new JButton("+"); ba.setBounds(420,110,70,30); c.add(ba); ba.setBackground(Color.BLACK); ba.setForegr
ound(Color.WHITE);
bs=new JButton("-"); bs.setBounds(500,110,70,30); c.add(bs); bs.setBackground(Color.BLACK); bs.setForegr
ound(Color.WHITE);
bm=new JButton("x"); bm.setBounds(420,150,70,30); c.add(bm); bm.setBackground(Color.BLACK); bm.setForegr
ound(Color.WHITE);
beq=new JButton("="); beq.setBounds(460,190,70,30); c.add(beq); beq.setBackground(Color.BLACK); beq.setF
oreground(Color.WHITE);
bc=new JButton("c"); bc.setBounds(260,190,70,30); c.add(bc); bc.setBackground(Color.BLACK); bc.setForegr
ound(Color.WHITE);
b1.addActionListener(this); b2.addActionListener(this); b3.addActionListener(this); b4.addActionListener
(this);
b5.addActionListener(this); b6.addActionListener(this); b7.addActionListener(this); b8.addActionListener
(this);
b9.addActionListener(this); b0.addActionListener(this); bc.addActionListener(this); ba.addActionListener
(this);
bs.addActionListener(this); bm.addActionListener(this); bd.addActionListener(this); beq.addActionListene
r(this);}
                                                                                                  25,1          27%
```

```
bs.addActionListener(this); bm.addActionListener(this); bd.addActionListener(this); beq.addActionListene
r(this);}
public void actionPerformed(ActionEvent e){int a=0; String t; if(e.getSource()==bc){t1.setText("");}
if(e.getSource()==b1){t=t1.getText(); a=t.length(); if(a==0){t1.setText("1");}
if(a>0 && a<4){t1.setText(t+"1");}}
if(e.getSource()==b2){t=t1.getText(); a=t.length(); if(a==0){t1.setText("2");}
if(a>0 && a<4){t1.setText(t+"2");}}
if(e.getSource()==b3){t=t1.getText(); a=t.length(); if(a==0){t1.setText("3");}
if(a>0 && a<4){t1.setText(t+"3");}}
if(e.getSource()==b4){t=t1.getText(); a=t.length(); if(a==0){t1.setText("4");}
if(a>0 && a<4){t1.setText(t+"4");}}
if(e.getSource()==b5){t=t1.getText(); a=t.length(); if(a==0){t1.setText("5");}
if(a>0 && a<4){t1.setText(t+"5");}}
if(e.getSource()==b6){t=t1.getText(); a=t.length(); if(a==0){t1.setText("6");}
if(a>0 && a<4){t1.setText(t+"6");}}
if(e.getSource()==b7){t=t1.getText(); a=t.length(); if(a==0){t1.setText("7");}
if(a>0 && a<4){t1.setText(t+"7");}}
if(e.getSource()==b8){t=t1.getText(); a=t.length(); if(a==0){t1.setText("8");}
if(a>0 && a<4){t1.setText(t+"8");}}
if(e.getSource()==b9){t=t1.getText(); a=t.length(); if(a==0){t1.setText("9");}
if(a>0 && a<4){t1.setText(t+"9");}}
if(e.getSource()==b0){t=t1.getText(); a=t.length(); if(a==0){t1.setText("0");}
if(a>0 && a<4){t1.setText(t+"0");}}
if(e.getSource()==ba){t=t1.getText(); a=t.length(); if(a>0){b=Integer.parseInt(t);
t1.setText(""); s=1;}}
if(e.getSource()==bs){t=t1.getText(); a=t.length(); if(a>0){b=Integer.parseInt(t);
                                                                                    32,1          64%
```

```
if(a>0 && a<4){t1.setText(t+"0");}}
if(e.getSource()==ba){t=t1.getText(); a=t.length(); if(a>0){b=Integer.parseInt(t);
t1.setText(""); s=1;}}
if(e.getSource()==bs){t=t1.getText(); a=t.length(); if(a>0){b=Integer.parseInt(t);
t1.setText(""); s=2;}}
if(e.getSource()==bm){t=t1.getText(); a=t.length(); if(a>0){b=Integer.parseInt(t);
t1.setText(""); s=3;}}
if(e.getSource()==bd){t=t1.getText(); a=t.length(); if(a>0){b=Integer.parseInt(t);
t1.setText(""); s=4;}}
if(e.getSource()==beq){t=t1.getText(); a=t.length(); if(a>0){d=Integer.parseInt(t);
if(s==1){f=b+d;} if(s==2){f=b-d;} if(s==3){f=b*d;} if(s==4){f=b/d;} t1.setText(""+f);}}
}}

public class App {

    public static void main(String[] args) {
        MyFrame frame=new MyFrame();
    }
}
~
~
~
~
~
~
~
                                                                                    54,1          Bot
```

Output:


Simple Calculator
1
2
3
4
+
-
5
6
7
8
x
/
9
0
c
=

Q36. Write two different programs to copy a file in another file by character by character and byte by byte methods.

Code:

```
// Java program to copy content from
// one file to another

import java.io.*;
import java.util.*;

public class App {

  public static void copyContent(File a, File b) throws Exception {
    FileInputStream in = new FileInputStream(a);
    FileOutputStream out = new FileOutputStream(b);

    try {
      int n;

      // read() function to read the
      // byte of data
      while ((n = in.read()) != -1) {
        // write() function to write
        // the byte of data
        out.write(n);
      }
    } finally {
      if (in != null) {
        // close() function to close the
        // stream
        in.close();
      }
      // close() function to close
      // the stream
      if (out != null) {
        out.close();
      }
    }
    System.out.println("File Copied");
```

```
  }

  public static void main(String[] args) throws Exception {
    Scanner sc = new Scanner(System.in);

    // get the source file name
    System.out.println(
      "Enter the source filename from where you have to read/copy :"
    );
    String a = sc.nextLine();

    // source file
    File x = new File(a);

    // get the destination file name
    System.out.println(
      "Enter the destination filename where you have to write/paste :"
    );
    String b = sc.nextLine();

    // destination file
    File y = new File(b);

    // method called to copy the
    // contents from x to y
    copyContent(x, y);
  }
}
```

```
import java.io.*;
import java.util.*;

public class App {

  public static void copyContent(File a, File b) throws Exception {
    FileInputStream in = new FileInputStream(a);
    FileOutputStream out = new FileOutputStream(b);

    try {
      int n;

      // read() function to read the
      // byte of data
      while ((n = in.read()) != -1) {
        // write() function to write
        // the byte of data
        out.write(n);
      }
    } finally {
      if (in != null) {
        // close() function to close the
        // stream
        in.close();
      }
      // close() function to close
                                                                    9,1          8%
```

```
      }
      // close() function to close
      // the stream
      if (out != null) {
        out.close();
      }
    }
    System.out.println("File Copied");
  }

  public static void main(String[] args) throws Exception {
    Scanner sc = new Scanner(System.in);

    // get the source file name
    System.out.println(
      "Enter the source filename from where you have to read/copy :"
    );
    String a = sc.nextLine();

    // source file
    File x = new File(a);

    // get the destination file name
    System.out.println(
      "Enter the destination filename where you have to write/paste :"
    );
                                                                    33,1         72%
```

```
    // get the destination file name
    System.out.println(
      "Enter the destination filename where you have to write/paste :"
    );
    String b = sc.nextLine();

    // destination file
    File y = new File(b);

    // method called to copy the
    // contents from x to y
    copyContent(x, y);
  }
}
~
~
~
~
~
~
~
~
~
~
~
                                                                    57,1          Bot
```

Output:

```
┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q36/src]
└─$ java App                                                                  34 ⚙
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Enter the source filename from where you have to read/copy :
/mnt/1A10343510341A69/sahil/my C language/java/Assignment/Q25/src/App.java
Enter the destination filename where you have to write/paste :
/home/kalikali/Documents/
Exception in thread "main" java.io.FileNotFoundException: /home/kalikali/Documents (Is a directory)
        at java.base/java.io.FileOutputStream.open0(Native Method)
        at java.base/java.io.FileOutputStream.open(FileOutputStream.java:295)
        at java.base/java.io.FileOutputStream.<init>(FileOutputStream.java:236)
        at java.base/java.io.FileOutputStream.<init>(FileOutputStream.java:185)
        at App.copyContent(App.java:11)
        at App.main(App.java:61)

┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q36/src]
└─$ java App                                                            1 × 34 ⚙
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Enter the source filename from where you have to read/copy :
/mnt/1A10343510341A69/sahil/my C language/java/Assignment/Q25/src/App.java
Enter the destination filename where you have to write/paste :
/home/kalikali/Documents/App.java
File Copied

┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q36/src]
└─$                                                                           34 ⚙
```

Q37. Write a Java application to print Pascal's triangle.

Code:

```
// Print Pascal's Triangle in Java
import java.io.*;

public class App {

  public int factorial(int i) {
    if (i == 0) return 1;
    return i * factorial(i - 1);
  }

  public static void main(String[] args) {
    int n = 4, i, j;
    App g = new App();
    for (i = 0; i <= n; i++) {
      for (j = 0; j <= n - i; j++) {
        // for left spacing
        System.out.print(" ");
      }
      for (j = 0; j <= i; j++) {
        // nCr formula
        System.out.print(
          " " + g.factorial(i) / (g.factorial(i - j) * g.factorial(j))
        );
      }

      // for newline
      System.out.println();
    }
  }
}
```

```
// Print Pascal's Triangle in Java
import java.io.*;

public class App {

  public int factorial(int i) {
    if (i == 0) return 1;
    return i * factorial(i - 1);
  }

  public static void main(String[] args) {
    int n = 4, i, j;
    App g = new App();
    for (i = 0; i <= n; i++) {
      for (j = 0; j <= n - i; j++) {
        // for left spacing
        System.out.print(" ");
      }
      for (j = 0; j <= i; j++) {
        // nCr formula
        System.out.print(
          " " + g.factorial(i) / (g.factorial(i - j) * g.factorial(j))
        );
      }

      // for newline
                                                                    12,1          Top
```

```
        // for left spacing
        System.out.print(" ");
      }
      for (j = 0; j <= i; j++) {
        // nCr formula
        System.out.print(
          " " + g.factorial(i) / (g.factorial(i - j) * g.factorial(j))
        );
      }

      // for newline
      System.out.println();
    }
  }
}
~
~
~
~
~
~
~
~
~
~
~
                                                                    21,1          Bot
```

Output:

```
┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q37/src]
└─$ java App                                                          35 ⚙
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
      1
     1 1
    1 2 1
   1 3 3 1
  1 4 6 4 1

┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q37/src]
└─$                                                                   35 ⚙
```

Q38. Write a program to compute the following series:

$1 – x + x^2/2! – x^3/3! + x^4/4! + ….. + (–/+ 1)nx^n/n!$

Where n and x is to be accepted by the user.

Code:

```
// Java implementation to find
// the sum of series
// 1 + x^2 + x^3 + ....+ x^n

public class App {

  // Java code to print the sum
  // of the given series
  static double sum(int x, int n) {
    double i, total = 1.0, multi = x;

    // First Term
    System.out.print("1 ");

    // Loop to print the N terms
    // of the series and compute
    // their sum
    for (i = 1; i < n; i++) {
      total = total + multi;
      System.out.print(multi);
      System.out.print(" ");
      multi = multi * x;
    }

    System.out.println();
    return total;
  }

  // Driver Code
  public static void main(String[] args) {
    int x = 2;
    int n = 5;
    System.out.printf("%.2f", sum(x, n));
```

```
  }
}
```

src : vim — Konsole

```
// Java implementation to find
// the sum of series
// 1 + x^2 + x^3 + ....+ x^n

public class App {

  // Java code to print the sum
  // of the given series
  static double sum(int x, int n) {
    double i, total = 1.0, multi = x;

    // First Term
    System.out.print("1 ");

    // Loop to print the N terms
    // of the series and compute
    // their sum
    for (i = 1; i < n; i++) {
      total = total + multi;
      System.out.print(multi);
      System.out.print(" ");
      multi = multi * x;
    }

    System.out.println();
    return total;
                                                    21,1          Top
```

src : vim — Konsole

```
    // of the series and compute
    // their sum
    for (i = 1; i < n; i++) {
      total = total + multi;
      System.out.print(multi);
      System.out.print(" ");
      multi = multi * x;
    }

    System.out.println();
    return total;
  }

  // Driver Code
  public static void main(String[] args) {
    int x = 2;
    int n = 5;
    System.out.printf("%.2f", sum(x, n));
  }
}
~
~
~
~
~
~
                                                    21,1          Bot
```

Output:

```
┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q38/src]
└─$ java App                                                                    36 ⚙
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
1 2.0 4.0 8.0 16.0
31.00

┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q38/src]
└─$                                                                             36 ⚙
```

Q39. Define a class BankAccount with appropriate member variables and methods to deposit and withdraw amount. Refine BankAccount to SavingBankAccount and CurrentBankAccount using inheritance. Use them in main class to demonstrate dynamic method dispatch.

Code:

```
import java.util.Scanner;

public class App {

  public static void main(String[] args) {
    Scanner in = new Scanner(System.in);

    App account = new App(1000);
    account.deposit(500);
    account.withdraw(50);

    System.out.println("BankAccount " + account.getNumber());
    System.out.println("Has a balance of " + account.getBalance());
  }

  // These are different for each account
  private double balance;
  private int accountNumber;

  // This is shared by all accounts, so it's static
  private static int lastAccountNumber = 0;

  // This is a constructor: no return type (void, boolean etc) and has the same name as the
class.
  public App(double intialBalance) {
    balance = intialBalance;

    accountNumber = lastAccountNumber + 1;
    lastAccountNumber = accountNumber;
  }

  public void deposit(double depositAmount) {
    balance += depositAmount;
  }
```

```
  public boolean withdraw(double withdrawAmount) {
    if (withdrawAmount > balance) {
      System.out.println("Insufficient Funds!!!");
      return false;
    } else {
      balance -= withdrawAmount;
      return true;
    }
  }

  public int getNumber() {
    return accountNumber;
  }

  public double getBalance() {
    return balance;
  }
}
```

```
import java.util.Scanner;

public class App {

  public static void main(String[] args) {
    Scanner in = new Scanner(System.in);

    App account = new App(1000);
    account.deposit(500);
    account.withdraw(50);

    System.out.println("BankAccount " + account.getNumber());
    System.out.println("Has a balance of " + account.getBalance());
  }

  // These are different for each account
  private double balance;
  private int accountNumber;

  // This is shared by all accounts, so it's static
  private static int lastAccountNumber = 0;

  // This is a constructor: no return type (void, boolean etc) and has the same name as the class.
  public App(double intialBalance) {
    balance = intialBalance;

                                                                                    21,1          Top
```

```
  // This is shared by all accounts, so it's static
  private static int lastAccountNumber = 0;

  // This is a constructor: no return type (void, boolean etc) and has the same name as the class.
  public App(double intialBalance) {
    balance = intialBalance;

    accountNumber = lastAccountNumber + 1;
    lastAccountNumber = accountNumber;
  }

  public void deposit(double depositAmount) {
    balance += depositAmount;
  }

  public boolean withdraw(double withdrawAmount) {
    if (withdrawAmount > balance) {
      System.out.println("Insufficient Funds!!!");
      return false;
    } else {
      balance -= withdrawAmount;
      return true;
    }
  }

                                                                                    30,0-1        69%
```

```
  public boolean withdraw(double withdrawAmount) {
    if (withdrawAmount > balance) {
      System.out.println("Insufficient Funds!!!");
      return false;
    } else {
      balance -= withdrawAmount;
      return true;
    }
  }

  public int getNumber() {
    return accountNumber;
  }

  public double getBalance() {
    return balance;
  }
}
~
~
~
~
~
~
~
                                                                    39,1          Bot
```

Output:

```
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
BankAccount 1
Has a balance of 1450.0

┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q39/src]
└─$                                                                    37 ⚙
```

Q40. Create an abstract class Shape and derived classes Rectangle and Circle from Shape class. Implement abstract method of shape class in Rectangle and Circle class. Shape class contains: origin (x, y) as data member, display() and area() as abstract methods. Circle class contains: radius as data member. Rectangle class contains: length and width. (Use Inheritance, overloading and overriding concept)

Code:

```
interface Shape{void input();void area();}
class Circle implements Shape{
    int r = 0;double pi = 3.14, ar = 0;
    @Override public void input(){r = 5;}
    @Override public void area(){ar = pi * r * r;System.out.println("Area of
circle:"+ar);}}
class Rectangle extends Circle{
    int l = 0, b = 0;double ar;
    public void input(){super.input();l = 6;b = 4;}
    public void area(){super.area();ar = l * b;
        System.out.println("Area of rectangle:"+ar);}}
public class App{public static void main(String[] args){
        Rectangle obj = new Rectangle();obj.input();obj.area();}}
```

App.java - Q40 - Visual Studio Code

File Edit Selection View Go Run Terminal Help

App.java

src > App.java > App > main(String[])

```java
interface Shape{void input();void area();}
class Circle implements Shape{
    int r = 0;double pi = 3.14, ar = 0;
    @Override public void input(){r = 5;}
    @Override public void area(){ar = pi * r * r;System.out.
    println("Area of circle:"+ar);}}
class Rectangle extends Circle{
    int l = 0, b = 0;double ar;
    public void input(){super.input();l = 6;b = 4;}
    public void area(){super.area();ar = l * b;
        System.out.println("Area of rectangle:"+ar);}}
Run | Debug
public class App{public static void main(String[] args){
        Rectangle obj = new Rectangle();obj.input();obj.area
        ();}}
```

OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

TERMINAL

```
┌──(kalikali㉿kali)-[/mnt/…/my C language/java/Assignment/Q40]
└─$  /usr/bin/env /usr/lib/jvm/java-11-openjdk-amd64/bin/java -cp /mnt/1A10343510341A69/sahil/my\ C\ language/java/Assignment/Q40/bin
 App
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Area of circle:78.5
Area of rectangle:24.0
┌──(kalikali㉿kali)-[/mnt/…/my C language/java/Assignment/Q40]
└─$
```

Ln 12, Col 65 Spaces: 4 UTF-8 LF {} Java Go Live Prettier

Q41. Create a package named as MyPackage with a class named as Calculate. The class should contain three methods with the following specifications:

- **Volume(): Accepts three double type arguments i.e. width, height, depth; calculate volume and return double type value.**
- **Add(): Accepts two integer type values, adds them and returns the value.**
- **Divide(): Accepts two integer type values, adds them and returns results.**

Import this package into a file named as PackageDemo and call the above three methods.

Code:

```
package MyPackage;

public class Calculate {
   double height,width,depth,c,d; int a,b;
   public double volumn(double height, double width, double depth){
      this.depth = depth; this.height=height; this.width=width;
      return depth*height*width;
   }
   public int add(int a,int b){
      this.a=a; this.b=b;
      return a+b;
   }
   public double divide(double c,double d){
      this.c=c; this.d=d;
      return c/d;
   }
}
package PackageDemo;

import MyPackage.Calculate;

public class App {
   public static void main(String[] args) throws Exception {
      Calculate calculate = new Calculate();
      System.out.println("The addition :: "+calculate.add(10, 40));
      System.out.println("The division :: "+calculate.divide(50, 10));
      System.out.println("The volumn of cubeboid :: "+calculate.volumn(25, 54, 60));
```

```
    }
}
```

MyPackage : vim — Konsole

```
package MyPackage;

public class Calculate {
    double height,width,depth,c,d; int a,b;
    public double volumn(double height, double width, double depth){
        this.depth = depth; this.height=height; this.width=width;
        return depth*height*width;
    }
    public int add(int a,int b){
        this.a=a; this.b=b;
        return a+b;
    }
    public double divide(double c,double d){
        this.c=c; this.d=d;
        return c/d;
    }
}
~
~
~
~
~
~
~
~
~
~
"Calculate.java" 17L, 453B                                    1,1           All
```

PackageDemo : vim — Konsole

```
package PackageDemo;

import MyPackage.Calculate;

public class App {
    public static void main(String[] args) throws Exception {
        Calculate calculate = new Calculate();
        System.out.println("The addition :: "+calculate.add(10, 40));
        System.out.println("The division :: "+calculate.divide(50, 10));
        System.out.println("The volumn of cubeboid :: "+calculate.volumn(25, 54, 60));
    }
}
~
~
~
~
~
~
~
~
~
~
~
~
~
"App.java" 12L, 417B                                                    1,1           All
```

Output:

App.java - Q41 - Visual Studio Code

File Edit Selection View Go Run Terminal Help

EXPLORER

Q41
- .vscode
- bin
- lib
- src
  - MyPackage
    - .Calculate.java.swp
    - Calculate.java
  - PackageDemo
    - .App.java.swp
    - App.java
- README.md

OUTLINE

TIMELINE

JAVA PROJECTS
- Q41
  - src
    - MyPackage
    - PackageDemo
      - App
  - JRE System Library [java-11-...
  - Referenced Libraries

Calculate.java | App.java

src > PackageDemo > App.java > App > main(String[])

```java
package PackageDemo;

import MyPackage.Calculate;

public class App {
    Run | Debug
    public static void main(String[] args) throws Exception {
        Calculate calculate = new Calculate();
        System.out.println("The addition :: "+calculate.add(a
        System.out.println("The division :: "+calculate.divide
        System.out.println("The volumn of cubeboid :: "+calcul
    }
}
```

OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

TERMINAL

```
└─$ /usr/bin/env /usr/lib/jvm/java-11-openjdk-amd64/bin/java -cp /mnt/1A10343510341A69/sahil/my\ C\ language/java/Assignment/Q41/bin PackageDemo.App
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
The addition :: 50
The division :: 5.0
The volumn of cubeboid :: 81000.0

┌──(kalikali㉿kali)-[/mnt/…/my C language/java/Assignment/Q41]
└─$
```

zsh

Run: App

Ln 10, Col 57 Spaces: 4 UTF-8 LF Java Go Live Prettier

Q42. Write a program that will count number of characters, words and lines in a file. The name should be passes as command line argument.

Code:

```
import java.util.Scanner;

public class App {
   public static void main(String[] args) throws Exception {
      new WordCounter();
   }
}
class WordCounter{
   int word,chars,spaces,lines;
   WordCounter(){
      System.out.print("Enter the sentence :: ");
      Scanner input = new Scanner(System.in);
      String n = input.nextLine();
      word += n.split("\\s").length;spaces += word-1;chars += n.length(); lines +=
n.split("\\n").length;
      System.out.println("The total words in the sentence :: "+word);
      System.out.println("The total characters in the sentence :: "+chars);
      System.out.println("The total spaces in the sentence :: "+spaces);
      System.out.println("The total lines in the sentence :: "+lines);

   }
}
```

```
import java.util.Scanner;

public class App {
    public static void main(String[] args) throws Exception {
        new WordCounter();
    }
}
class WordCounter{
    int word,chars,spaces,lines;
    WordCounter(){
        System.out.print("Enter the sentence :: ");
        Scanner input = new Scanner(System.in);
        String n = input.nextLine();
        word += n.split("\\s").length;spaces += word-1;chars += n.length(); lines += n.split("\\n").length;
        System.out.println("The total words in the sentence :: "+word);
        System.out.println("The total characters in the sentence :: "+chars);
        System.out.println("The total spaces in the sentence :: "+spaces);
        System.out.println("The total lines in the sentence :: "+lines);

    }
}
~
~
~
~
~
~
                                                                    21,1          All
```

Output:

```
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Enter the sentence :: i am a student of bca last year
The total words in the sentence :: 8
The total characters in the sentence :: 31
The total spaces in the sentence :: 7
The total lines in the sentence :: 1

┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q42/src]
└─$                                                                    40 ⚙
```

Q43. Write a multi-threaded Java program to print all numbers below 100,000 that are both prime and Fibonacci number (some examples are 2, 3, 5, 13, etc.). Design a thread that generates prime numbers below 100,000 and writes them into a pipe. Design another thread that generates Fibonacci numbers and writes them to another pipe. The main thread should read both the pipes to identify numbers common to both.

Code:

```
import java.util.*;
import java.io.*;
class Fibonacci extends Thread{
   private PipedWriter out = new PipedWriter();
   public PipedWriter getPipedWriter(){
      return out;
   }
   public void run(){
      Thread t = Thread.currentThread();t.setName("Fibonacci");
      System.out.println(t.getName() + " thread started");
      int fibo1=0,fibo2=1,fibo=0;
      while(true){
         try{
            fibo = fibo1 + fibo2;
            if(fibo>100000){
               out.close();break;
            }out.write(fibo);sleep(100);
         }
         catch(Exception e){
            System.out.println("Fibonacci:"+e);
         }
         fibo1=fibo2;fibo2=fibo;
      }
      System.out.println(t.getName() + " thread exiting");
   }
}
class Prime extends Thread
{
   private PipedWriter out1 = new PipedWriter();
   public PipedWriter getPipedWriter(){
      return out1;
```

```
    }
    public void run(){
        Thread t= Thread.currentThread();t.setName("Prime");
        System.out.println(t.getName() + " thread Started...");
        int prime=1;
        while(true){
            try{
                if(prime>100000){out1.close();break;}
                if(isPrime(prime))
                out1.write(prime);prime++;sleep(0);
            }
            catch(Exception e){
                System.out.println(t.getName() + " thread exiting.");
                System.exit(0);
            }
        }
    }public boolean isPrime(int n){
        int m=(int)Math.round(Math.sqrt(n));
        if(n==1 || n==2)
        return true;
        for(int i=2;i<=m;i++)
        if(n%i==0)
        return false;
        return true;
    }
}
public class App{
    public static void main(String[] args) throws Exception    {
        Thread t=Thread.currentThread();t.setName("Main");
        System.out.println(t.getName() + " thread Started...");
        Fibonacci fibonacci = new Fibonacci();Prime prime = new Prime();
        PipedReader fpr = new PipedReader(fibonacci.getPipedWriter());
        PipedReader ppr = new PipedReader(prime.getPipedWriter());
        fibonacci.start();prime.start();
        int fib=fpr.read(), prm=ppr.read();
        System.out.println("The numbers common to PRIME and FIBONACCI:");
        while((fib!=-1) && (prm!=-1)){
            while(prm<=fib){
```

```
            if(fib==prm)
            System.out.println(prm);
            prm=ppr.read();
        }fib=fpr.read();
    }System.out.println(t.getName() + " thread exiting");
  }
}
```

src : vim — Konsole

```
import java.util.*;
import java.io.*;
class Fibonacci extends Thread{
    private PipedWriter out = new PipedWriter();
    public PipedWriter getPipedWriter(){
        return out;
    }
    public void run(){
        Thread t = Thread.currentThread();t.setName("Fibonacci");
        System.out.println(t.getName() + " thread started");
        int fibo1=0,fibo2=1,fibo=0;
        while(true){
            try{
                fibo = fibo1 + fibo2;
                if(fibo>100000){
                    out.close();break;
                }out.write(fibo);sleep(100);
            }
            catch(Exception e){
                System.out.println("Fibonacci:"+e);
            }
            fibo1=fibo2;fibo2=fibo;
        }
        System.out.println(t.getName() + " thread exiting");
                                                                    19,1          Top
```

src : vim — Konsole

```
            fibo1=fibo2;fibo2=fibo;
        }
        System.out.println(t.getName() + " thread exiting");
    }
}
class Prime extends Thread
{
    private PipedWriter out1 = new PipedWriter();
    public PipedWriter getPipedWriter(){
        return out1;
    }
    public void run(){
        Thread t= Thread.currentThread();t.setName("Prime");
        System.out.println(t.getName() + " thread Started... ");
        int prime=1;
        while(true){
            try{
                if(prime>100000){out1.close();break;}
                if(isPrime(prime))
                out1.write(prime);prime++;sleep(0);
            }
            catch(Exception e){
                System.out.println(t.getName() + " thread exiting.");
                System.exit(0);
                                                                30,1          40%
```

src : vim — Konsole

```
            catch(Exception e){
                System.out.println(t.getName() + " thread exiting.");
                System.exit(0);
            }
        }
    }public boolean isPrime(int n){
        int m=(int)Math.round(Math.sqrt(n));
        if(n==1 || n==2)
        return true;
        for(int i=2;i<=m;i++)
        if(n%i==0)
        return false;
        return true;
    }
}
public class App{
    public static void main(String[] args) throws Exception    {
        Thread t=Thread.currentThread();t.setName("Main");
        System.out.println(t.getName() + " thread Started... ");
        Fibonacci fibonacci = new Fibonacci();Prime prime = new Prime();
        PipedReader fpr = new PipedReader(fibonacci.getPipedWriter());
        PipedReader ppr = new PipedReader(prime.getPipedWriter());
        fibonacci.start();prime.start();
        int fib=fpr.read(), prm=ppr.read();
                                                                    48,1          80%
```

src : vim — Konsole

```
public class App{
    public static void main(String[] args) throws Exception    {
        Thread t=Thread.currentThread();t.setName("Main");
        System.out.println(t.getName() + " thread Started ... ");
        Fibonacci fibonacci = new Fibonacci();Prime prime = new Prime();
        PipedReader fpr = new PipedReader(fibonacci.getPipedWriter());
        PipedReader ppr = new PipedReader(prime.getPipedWriter());
        fibonacci.start();prime.start();
        int fib=fpr.read(), prm=ppr.read();
        System.out.println("The numbers common to PRIME and FIBONACCI:");
        while((fib≠-1) && (prm≠-1)){
            while(prm≤fib){
                if(fib==prm)
                System.out.println(prm);
                prm=ppr.read();
            }fib=fpr.read();
        }System.out.println(t.getName() + " thread exiting");
    }
}
~
~
~
~
~
                                                    66,1          Bot
```

Output:

```
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Main thread Started...
Fibonacci thread started
Prime thread Started...
The numbers common to PRIME and FIBONACCI:
1
2
3
5
13
89
233
1597
Fibonacci thread exiting
28657
Main thread exiting
Prime thread exiting.

┌──(kalikali㉿kali)-[/mnt/…/java/Assignment/Q43/src]
└─$                                                                    41 ⚙
```