Class Work - 1

Q. Write a "Hello World" Program.

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

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\Hello.java

Hello World!

Q. Write a program using Command Line Arguments.

```
public class CommandLine {
    public static void main(String args[])
    {
        System.out.println("Argument is : " + args[0]);
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\CommandLine.java 45

Argument is : 45

Q. Write a program for Arithmetic Calculations.

```java
public class Arithmetic {
    public static void main(String args[])
    {
        System.out.println("Addition = " + (Integer.parseInt(args[0])
+ Integer.parseInt(args[1])) );
        System.out.println("Subtraction = " + (Integer.parseInt(args[0
]) - Integer.parseInt(args[1])) );
        System.out.println("Multiplication = " + (Integer.parseInt(arg
s[0]) * Integer.parseInt(args[1])) );
        System.out.println("Division = " + (Integer.parseInt(args[0])
/ Integer.parseInt(args[1])) );
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\Arithmetic.java 3 6

Addition = 9

Subtraction = -3

Multiplication = 18

Division = 0

Q. Write a Program for Floating Point Numbers.

```java
public class FloatingPoints {
    public static void main(String[] args)
    {
        int m = 4;
        double n = 2;
        double x = 3.5;
        long k = 10L;
        System.out.println((n+x)*m + k);
    }
}
```

PS E:\MCA\MCA SEM 3\JAVA> java .\FloatingPoints.java

32.0

Q. Write a Program for GallonToLitres.

> ## GalToLit

```java
public class GalToLit {
    public static void main(String args[])
    {
        double gallons,litres;
        gallons = 45;
        litres = gallons * 3.7854;
        System.out.println(gallons + " Gallons is " + litres + " litre
s.");
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\GalToLit.java

45.0 Gallons is 170.34300000000002 litres.

> ## GalToLit2

```java
public class GalToLit2 {
    public static void main(String args[])
    {
        double gallons, liters;
        int counter;
        counter = 0;
        for(gallons = 1; gallons < 100; gallons++)
        {
            liters = gallons * 3.7854;
            System.out.println(gallons + " Gallons is " + liters + " l
itres.");
            counter++;
            if(counter == 10)
            {
```

```java
            System.out.println();
            counter = 0;
        }
    }
}
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\GalToLit2.java

1.0 Gallons is 3.7854 litres.

2.0 Gallons is 7.5708 litres.

3.0 Gallons is 11.356200000000001 litres.

4.0 Gallons is 15.1416 litres.

5.0 Gallons is 18.927 litres.

6.0 Gallons is 22.712400000000002 litres.

7.0 Gallons is 26.4978 litres.

8.0 Gallons is 30.2832 litres.

9.0 Gallons is 34.0686 litres.

10.0 Gallons is 37.854 litres.

11.0 Gallons is 41.6394 litres.

12.0 Gallons is 45.424800000000005 litres.

13.0 Gallons is 49.2102 litres.

14.0 Gallons is 52.9956 litres.

15.0 Gallons is 56.781 litres.

16.0 Gallons is 60.5664 litres.

17.0 Gallons is 64.3518 litres.

18.0 Gallons is 68.1372 litres.

19.0 Gallons is 71.9226 litres.

20.0 Gallons is 75.708 litres.

21.0 Gallons is 79.49340000000001 litres.

22.0 Gallons is 83.2788 litres.

23.0 Gallons is 87.0642 litres.

24.0 Gallons is 90.84960000000001 litres.

25.0 Gallons is 94.635 litres.

26.0 Gallons is 98.4204 litres.

27.0 Gallons is 102.2058 litres.

28.0 Gallons is 105.9912 litres.

29.0 Gallons is 109.7766 litres.

30.0 Gallons is 113.562 litres.

31.0 Gallons is 117.34740000000001 litres.

32.0 Gallons is 121.1328 litres.

33.0 Gallons is 124.9182 litres.

34.0 Gallons is 128.7036 litres.

35.0 Gallons is 132.489 litres.

36.0 Gallons is 136.2744 litres.

37.0 Gallons is 140.0598 litres.

38.0 Gallons is 143.8452 litres.

39.0 Gallons is 147.63060000000002 litres.

40.0 Gallons is 151.416 litres.

41.0 Gallons is 155.2014 litres.

42.0 Gallons is 158.98680000000002 litres.

43.0 Gallons is 162.7722 litres.

44.0 Gallons is 166.5576 litres.

45.0 Gallons is 170.34300000000002 litres.

46.0 Gallons is 174.1284 litres.

47.0 Gallons is 177.9138 litres.

48.0 Gallons is 181.69920000000002 litres.

49.0 Gallons is 185.4846 litres.

50.0 Gallons is 189.27 litres.

51.0 Gallons is 193.0554 litres.

52.0 Gallons is 196.8408 litres.

53.0 Gallons is 200.6262 litres.

54.0 Gallons is 204.4116 litres.

55.0 Gallons is 208.197 litres.

56.0 Gallons is 211.9824 litres.

57.0 Gallons is 215.7678 litres.

58.0 Gallons is 219.5532 litres.

59.0 Gallons is 223.3386 litres.

60.0 Gallons is 227.124 litres.

61.0 Gallons is 230.9094 litres.

62.0 Gallons is 234.69480000000001 litres.

63.0 Gallons is 238.4802 litres.

64.0 Gallons is 242.2656 litres.

65.0 Gallons is 246.05100000000002 litres.

66.0 Gallons is 249.8364 litres.

67.0 Gallons is 253.6218 litres.

68.0 Gallons is 257.4072 litres.

69.0 Gallons is 261.1926 litres.

70.0 Gallons is 264.978 litres.

71.0 Gallons is 268.7634 litres.

72.0 Gallons is 272.5488 litres.

73.0 Gallons is 276.3342 litres.

74.0 Gallons is 280.1196 litres.

75.0 Gallons is 283.90500000000003 litres.

76.0 Gallons is 287.6904 litres.

77.0 Gallons is 291.4758 litres.

78.0 Gallons is 295.26120000000003 litres.

79.0 Gallons is 299.0466 litres.

80.0 Gallons is 302.832 litres.

81.0 Gallons is 306.61740000000003 litres.

82.0 Gallons is 310.4028 litres.

83.0 Gallons is 314.1882 litres.

84.0 Gallons is 317.97360000000003 litres.

85.0 Gallons is 321.759 litres.

86.0 Gallons is 325.5444 litres.

87.0 Gallons is 329.32980000000003 litres.

88.0 Gallons is 333.1152 litres.

89.0 Gallons is 336.9006 litres.

90.0 Gallons is 340.68600000000004 litres.

91.0 Gallons is 344.4714 litres.

92.0 Gallons is 348.2568 litres.

93.0 Gallons is 352.04220000000004 litres.

94.0 Gallons is 355.8276 litres.

95.0 Gallons is 359.613 litres.

96.0 Gallons is 363.39840000000004 litres.

97.0 Gallons is 367.1838 litres.

98.0 Gallons is 370.9692 litres.

99.0 Gallons is 374.7546 litres.

Q. Write a Program for If Statement Demo.

```java
public class IfDemo {
    public static void main(String args[])
    {
        int a, b, c;
        a = 2;
        b = 3;
        if(a < b)
            System.out.println("A is Smaller \n");

        if(a == b)
            System.out.println("A And B is equal \n");

        c = a - b;
        System.out.println("C contains " + c);

        if(c >= 0)
            System.out.println("C is Non-Negative \n");

        if(c < 0)
            System.out.println("C is Negative \n");

        c = b - a;
        System.out.println("C contains " + c);

        if(c >= 0)
            System.out.println("C is Non-Negative \n");

        if(c < 0)
            System.out.println("C is Negative \n");
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\IfDemo.java

A is Smaller

C contains -1

C is Negative

C contains 1

C is Non-Negative

Q. Write a Program To Find Max, Total & Display list.

```java
public class MSD {
    public static void main(String[] args)
    {
        int[] numbers = {34, 21, 56, 89, 12};
        int count = 0;
        int max = numbers[0];
        int total = 0;
        System.out.print("Numbers In Array : ");
        while(count < 5)
        {
            if( numbers[count] > max && count > 0)
                max = numbers[count];

            total += numbers[count];
            System.out.print(numbers[count++] + " ");
        }
        System.out.println("\nMax Element In Array : " + max);
        System.out.println("Total of Elements In Array : " + total);
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\MSD.java

Numbers In Array : 34 21 56 89 12

Max Element In Array : 89

Total of Elements In Array : 212

Q. Write a Program Character Arithmetic Demo.

```java
public class CharArithmetic {
    public static void main(String args[])
    {
        char symbol = 'A';
        System.out.println(symbol);
        ++symbol;
        System.out.println(symbol);
        symbol = 78;
        System.out.println(symbol);
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\CharArithmetic.java

A

B

N

Q. Write a Program for Sound Demo.

```java
public class Sound {
    public static void main(String args[])
    {
        double dist;
        dist = 11.2 * 1100;
        System.out.println("The Lighting is " + dist + " feet away.");
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\Sound.java

The Lighting is 12320.0 feet away.

Q. Write a Program for Scope Demo.

```java
public class ScopeDemo {
    public static void main(String args[])
    {
        int x;
        x = 10;
        if(x == 10)
        {
            int y = 20;
            System.out.println("X and Y : " + x + " " + y);
            x = y * 2;
        }
        // y = 100; //<-- Error Y is outside of its scope.
        System.out.println("X : " + x); // while x still known here.
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\ScopeDemo.java

X and Y : 10 20

X : 40

Q. Write a Program for Boolean.

```java
public class Boolean {
    public static void main(String args[])
    {
        boolean p, q;
        System.out.println("P\tQ\tAND\tOR\tNOT");
        p = true;
        q = true;
        System.out.print(p + "\t" + q + "\t");
        System.out.print((p&q) + "\t" + (p|q) + "\t");
        System.out.println((p^q) + "\t" + (!p));

        p = true;
        q = false;
        System.out.print(p + "\t" + q + "\t");
        System.out.print((p&q) + "\t" + (p|q) + "\t");
        System.out.println((p^q) + "\t" + (!p));

        p = false;
        q = true;
        System.out.print(p + "\t" + q + "\t");
        System.out.print((p&q) + "\t" + (p|q) + "\t");
        System.out.println((p^q) + "\t" + (!p));
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\Boolean.java

P   Q   AND   OR   NOT

true   true   true   true   false   false

true   false   false   true   true   false

false   true   false   true   true   true

Q. Write a Program for Mutiplication Table.

```java
public class Tables {
    public static void main(String args[])
    {
        for(int i = 1; i <= Integer.parseInt(args[1]); i++)
        {
            System.out.println(args[0] + " * " + i + " = " + Integer.p
arseInt(args[0]) * i);
        }
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\Tables.java 17 12

17 * 1 = 17

17 * 2 = 34

17 * 3 = 51

17 * 4 = 68

17 * 5 = 85

17 * 6 = 102

17 * 7 = 119

17 * 8 = 136

17 * 9 = 153

17 * 10 = 170

17 * 11 = 187

17 * 12 = 204

Q. Write a Program for Bit Demo.

```java
public class BitDemo {
    public static void main(String[] args)
    {
        int bitmask =0x000F;
        int val = 0x2222;
        System.out.println(val & bitmask);
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\BitDemo.java

2

Q. Write a Program for MathClass.

```java
public class MathClass {
    public static void main(String[] args) {

        int fishCount = 20;
        int fishLength = 10;
        int lengthPerSqFt = 2;
        double radius = 0.0;
        int feet = 0;
        int inches = 0;
        double pondArea = (double)(fishCount*fishLength)/lengthPerSqFt;

        radius = Math.sqrt(pondArea/Math.PI);
        feet = (int)Math.floor(radius);
        inches = (int)Math.round(12.0*(radius - feet));

        System.out.println("To hold " + fishCount + " fish averaging "
 + fishLength + " inches long you need a pond with an area of " + pond
Area + " square feet.");

        System.out.println("The radius of a pond with area " + pondAre
a + " square feet is " + feet + " feet " + inches + " inches");
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\MathClass.java

To hold 20 fish averaging 10 inches long you need a pond with an area of 100.0 square feet.

The radius of a pond with area 100.0 square feet is 5 feet 8 inches

Q. Arithmetic with Character Code.

```java
public class CharArithmetic {
    public static void main(String args[])
    {
        char symbol = 'A';
        System.out.println(symbol);
        ++symbol;
        System.out.println(symbol);
        symbol = 78;
        System.out.println(symbol);
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\CharArithmetic.java

A

B

N

Q. Shift Operators.

```java
import static java.lang.Long.toHexString;
public class ShiftOperator {
    public static void main(String[] args) {
        char letterA = 'A';
        char letterB = 'B';
        char letterC = 'C';
        char letterD = 'D';
        long packed = 0L;
        packed = letterD;
        packed = (packed << 16) | letterC;
        packed = (packed << 16) | letterB;
        packed = (packed << 16) | letterA;
        System.out.println("packed now contains 0x" + toHexString(packed));

        long mask = 0xFFFF;
        char letter = (char)(packed & mask);
        System.out.println("From right to left the letters in packed are:");
        System.out.println(" " + letter + " 0x" + toHexString(letter));

        packed >>= 16;
        letter = (char)(packed & mask);
        System.out.println(" " + letter + " 0x" + toHexString(letter));

        packed >>= 16;
        letter = (char)(packed & mask);
        System.out.println(" " + letter + " 0x" + toHexString(letter));

        packed >>= 16;
        letter = (char)(packed & mask);
        System.out.println(" " + letter + " 0x" + toHexString(letter));
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\ShiftOperator.java

packed now contains 0x44004300420041

From right to left the letters in packed are:

A 0x41

B 0x42

C 0x43

D 0x44

Q. Bitwise Operators (Indicators & Masking).

```java
import static java.lang.Integer.toBinaryString;
public class BitwiseOperator {
    public static void main(String[] args) {
        int indicators = 0xFF07;
        int selectBit3 = 0x4;
        System.out.println("indicators = " + toBinaryString(indicators
));
        System.out.println("selectBit3 = " + toBinaryString(selectBit3
));
        indicators &= selectBit3;
        System.out.println("indicators & selectBit3 = " + toBinaryStri
ng(indicators));
        indicators = 0xFF09;
        System.out.println("\nindicators = " + toBinaryString(indicato
rs));
        System.out.println("selectBit3 = " + toBinaryString(selectBit3
));
        indicators |= selectBit3;
        System.out.println("indicators | selectBit3 = " + toBinaryStri
ng(indicators));
        indicators &= ~selectBit3;
        System.out.println("\nThe third bit in the previous value of i
ndicators" + " has been switched off");
        System.out.println("indicators & ~selectBit3 = " + toBinaryStr
ing(indicators));
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\BitwiseOperator.java

indicators = 1111111100000111

selectBit3 = 100

indicators & selectBit3 = 100

indicators = 1111111100001001

selectBit3 = 100

indicators | selectBit3 = 1111111100001101

The third bit in the previous value of indicators has been switched off

indicators & ~selectBit3 = 1111111100001001

Q. Methods for operation on bits.

```java
import static java.lang.Long.*;
public class OperationOnBits {
    public static void main(String[] args) {
        long number = 0xF00000000000000FL;
        System.out.println("Number : " + toBinaryString(number));
        long result = rotateLeft(number,2);
        System.out.println("Number rotated left 2 bits : " + toBinaryS
tring(result));
        result = rotateRight(number, 3);
        System.out.println("Number rotated right 3 bits : " + toBinary
String(result));
        result = reverse(result);
        System.out.println("Previous result reversed : " + toBinaryStr
ing(result));
        System.out.println("Bit count in number : " + bitCount(number)
);
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\OperationOnBits.java

Number :
1111000000000000000000000000000000000000000000000000000000000000
1111

Number rotated left 2 bits :
1100000000000000000000000000000000000000000000000000000000000011
1111

Number rotated right 3 bits :
11111110000000000000000000000000000000000000000000000000000000000
0001

Previous result reversed :
10000000000000000000000000000000000000000000000000000000000000111
1111

Bit count in number : 8

Q. Deciphering Characters.

➔ Hard Way

```java
public class Deciphering1{
    public static void main(String[] args) {
        char symbol = 'A';
        symbol = (char)(128.0*Math.random());
        if(symbol >= 'A') {
            if(symbol <= 'Z') {
                System.out.println("You have the capital letter " + sy
mbol);
            }
            else {
                if(symbol >= 'a') {
                    if(symbol <= 'z') {
                        System.out.println("You have the small letter
" + symbol);
                    } else {
                        System.out.println("The code is greater than a
 but it's not a letter");
                    }
                } else {
                    System.out.println("The code is less than a and it
's not a letter");
                }
            }
        } else {
            System.out.println("The code is less than A so it's not a
letter");
        }
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\Deciphering1.java

You have the capital letter V

➔ **Easy Way**

```java
public class Deciphering2 {
    public static void main(String[] args) {
        char symbol = 'A';
        symbol = (char)(128.0*Math.random());
        if(symbol >= 'A' && symbol <= 'Z') {
            System.out.println("You have the capital letter " + symbol
);
        }
        else {
            if(symbol >= 'a' && symbol <= 'z') {
                System.out.println("You have the small letter " + symb
ol);
            }
            else {
                System.out.println("The code is not a letter");
            }
        }
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\Deciphering2.java

You have the small letter z

➜ Trivially

```java
import static java.lang.Character.isLowerCase;
import static java.lang.Character.isUpperCase;
public class Deciphering3 {
    public static void main(String[] args) {
        char symbol = 'A';
        symbol = (char)(128.0*Math.random()); // Generate a random cha
racter
        if(isUpperCase(symbol)) {
            System.out.println("You have the capital letter " + symbol
);
        } else {
            if(isLowerCase(symbol)) {
                System.out.println("You have the small letter " + symb
ol);
            } else {
                System.out.println("The code is not a letter");
            }
        }
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\Deciphering3.java

You have the small letter y

Q. Ternary Operator.

```java
public class Ternary {
    public static void main(String[] args) {
        int a = 10;
        boolean x = false;
        x = (a == 10) ? true : false;
        System.out.println(x);
        x = (a == 11) ? true : false;
        System.out.println(x);
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\Ternary.java

true

false

Q. Switch Case.

```java
public class SwitchCase {
    enum WashChoice {cotton, linen, wool, synthetic}
    public static void main(String[] args) {
        WashChoice wash = WashChoice.cotton;
        int clothes = 1;
        switch(clothes) {
            case 1:
            System.out.println("Washing shirts.");
            wash = WashChoice.cotton;
            break;
            case 2:
            System.out.println("Washing sweaters.");
            wash = WashChoice.wool;
            break;
            case 3:
            System.out.println("Washing socks.");
            wash = WashChoice.wool;
            break;
            case 4:
            System.out.println("Washing sheets.");
            wash = WashChoice.linen;
            break;
            case 5:
            System.out.println("Washing pants.");
            wash = WashChoice.synthetic;
            break;
            default:
            System.out.println("Unknown washing -
 default synthetic.");
            wash = WashChoice.synthetic;
            break;
        }
        System.out.println("Wash is "+ wash);
        switch(wash) {
            case wool:
```

```java
            System.out.println("Temperature is 120.");
            break;
        case cotton:
            System.out.println("Temperature is 170.");
            break;
        case synthetic:
            System.out.println("Temperature is 130.");
            break;
        case linen:
            System.out.println("Temperature is 180.");
            break;
        }
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\SwitchCase.java

Washing shirts.

Wash is cotton

Temperature is 170.

Q. Demonstrate Numeric for loop.

```java
public class ForLoop {
    public static void main(String[] args) {
        int limit = 20; // Sum from 1 to this value
        int sum = 0; // Accumulate sum in this variable
        // Loop from 1 to the value of limit, adding 1 each cycle
        for(int i = 1; i <= limit; i++)
        {
            sum += i;
            System.out.println("sum = " + sum);
        }
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\ForLoop.java

sum = 1

sum = 3

sum = 6

sum = 10

sum = 15

sum = 21

sum = 28

sum = 36

sum = 45

sum = 55

sum = 66

sum = 78

sum = 91

sum = 105

sum = 120

sum = 136

sum = 153

sum = 171

sum = 190

sum = 210

Q. Demonstrate Collection-Based for loop.

```java
public class CollectionForLoop {
    enum Season { spring, summer, monsoon, winter } // Enumeration type definition
    public static void main(String[] args) {
        for(Season season : Season.values()) { // Vary over all values
            System.out.println(" The season is now " + season);
        }
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\CollectionForLoop.java

 The season is now spring

 The season is now summer

 The season is now monsoon

 The season is now winter

Q. Demonstrate While loop.

```java
public class WhileLoop {
    public static void main(String[] args) {
        int limit = 20; // Sum from 1 to this value
        int sum = 0; // Accumulate sum in this variable
        int i = 1; // Loop counter
        // Loop from 1 to the value of limit, adding 1 each cycle
        while(i <= limit) {
            sum += i++; // Add the current value of i to sum
        }
        System.out.println("sum = " + sum);
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\WhileLoop.java

sum = 210

Q. Demonstrate Do While loop.

```java
public class DoWhileLoop {
    public static void main(String[] args) {
        int limit = 20; // Sum from 1 to this value
        int sum = 0; // Accumulate sum in this variable
        int i = 1; // Loop counter
        // Loop from 1 to the value of limit, adding 1 each cycle
        do {
            sum += i; // Add the current value of i to sum
            i++;
        } while(i <= limit);
        System.out.println("sum = " + sum);
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\DoWhileLoop.java

sum = 210

Q. Calculate Factorial.

```java
public class Factorial {
    public static void main(String[] args) {
        long limit = 20L; // Calculate factorials of integers up to th
is value
        long factorial = 1L; // A factorial will be stored in this var
iable
        // Loop from 1 to the value of limit
        for (long i = 1L; i <= limit; i++) {
            factorial = 1L; // Initialize factorial
            for (long factor = 2; factor <= i; factor++) {
                factorial *= factor;
            }
            System.out.println(i + "! is " + factorial);
        }
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\Factorial.java

1! is 1

2! is 2

3! is 6

4! is 24

5! is 120

6! is 720

7! is 5040

8! is 40320

9! is 362880

10! is 3628800

11! is 39916800

12! is 479001600

13! is 6227020800

14! is 87178291200

15! is 1307674368000

16! is 20922789888000

17! is 355687428096000

18! is 6402373705728000

19! is 121645100408832000

20! is 2432902008176640000

Q. Labled continue example.

```java
public class LabeledContinue {
    public static void main(String[] args) {
        long limit = 20L; // to calculate factorial of integers up to
this value
        long factorial = 1L; // factorial will be calculated in this v
ariable
        // Loop from 1 to the value of limit
        OuterLoop:
        for(long i = 1L; i <= limit; i++) {
            factorial = 1; // Initialize factorial
            for(long j = 2L; j <= i; j++)
            {
                if(i > 10L && i % 2L == 1L)
                {
                    continue OuterLoop; // Transfer to the outer loop
                }
                factorial *= j;
            }
            System.out.println(i + "! is " + factorial);
        }
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\LabeledContinue.java

1! is 1

2! is 2

3! is 6

4! is 24

5! is 120

6! is 720

7! is 5040

8! is 40320

9! is 362880

10! is 3628800

12! is 479001600

14! is 87178291200

16! is 20922789888000

18! is 6402373705728000

20! is 2432902008176640000

_____

Q. Calculating Primes.

> Prime - 1

```java
public class Primes {
    public static void main(String[] args) {
        int nValues = 50; // The maximum value to be checked
        boolean isPrime = true; // Is true if we find a prime
        // Check all values from 2 to nValues
        for(int i = 2; i <= nValues; i++) {
            isPrime=true; // Assume the current i is prime
            // Try dividing by all integers from 2 to i-1
            for(int j = 2; j < i; j++) {
                if(i % j == 0) { // This is true if j divides exactly
                    isPrime = false; // If we got here, it was an exact division
                    break; // so exit the loop
                }
            }
            // We can get here through the break, or through completing the loop
            if(isPrime) // So is it prime?
                System.out.println(i); // Yes, so output the value
        }
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\Primes.java

2

3

5

7

11

13

17

19

23

29

31

37

41

43

47

> **Prime - 2**

```java
public class Prime2 {
    public static void main(String[] args) {
        int nValues = 50; // The maximum value to be checked
        // Check all values from 2 to nValues
        OuterLoop:
        for(int i = 2; i <= nValues; i++) {
            // Try dividing by all integers from 2 to i-1
            for(int j = 2; j < i; j++) {
                if(i%j == 0) { // This is true if j divides exactly
                    continue OuterLoop; // so exit the loop
                }
            }
            // We only get here if we have a prime
            System.out.println(i); // so output the value
        }
    }
}
```

<u>Output :</u>

PS E:\MCA\MCA SEM 3\JAVA> java .\Prime2.java

2

3

5

7

11

13

17

19

23

29

31

37

41

43

47

Excercise - 1

Q. Exercise 4 question.

- **Write a program to display a random choice from a set of six choices for breakfast (you could use any set; for example, scrambled eggs, waffles, fruit, cereal, toast, or yogurt).**

```java
public class RandomBreakfast {
    enum breakfast{Scrambled_Eggs, Waffles, Fruits, Cereal, Toast, Yogurt}

    public static void main(String args[]){

        int choice = (int) (Math.random()*6);

        System.out.println(breakfast.values()[choice]);

    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\RandomBreakfast.java

Yogurt

- **When testing whether an integer is a prime, it is sufficient to try to divide by integers up to the square root of the number being tested. Rewrite the program example from this chapter to use this approach**

```java
class CheckPrime {
    public static void main(String args[])
    {
        boolean flag = false;
        int number = Integer.parseInt(args[0]);
        int sqroot = (int)Math.sqrt(number);

        for(int i = 2; i <= sqroot; i++)
        {
            if(number % i == 0)
            {
                flag = true;
                break;
            }
        }
        if(!flag)
            System.out.println(number + " a Prime Number.");

        else
            System.out.println(number + " is not a Prime Number.");
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\CheckPrime.java 17

17 a Prime Number.

PS E:\MCA\MCA SEM 3\JAVA> java .\CheckPrime.java 10

10 is not a Prime Number.

- **A lottery requires that you select six different numbers from the integers 1 to 49. Write a program to do this for you and generate five sets of entries.**

```java
public class Lottery {
    public static void main(String args[])
    {
        int [] numbers = new int[5];
        int random_number;
        boolean flag;


        for(int i = 0; i < 5; i++)
        {
            flag = false;
            random_number = (int) (Math.random()*49);
            for(int j = 0; j < i; j++)
            {
                if(numbers[j] == random_number)
                    flag = true;
            }

            if(!flag && random_number != 0){
                numbers[i] = random_number;
                System.out.println(numbers[i]);
            }
            else
                --i;

        }
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\Lottery.java

4

3

25

45

43

PS E:\MCA\MCA SEM 3\JAVA> java .\Lottery.java

31

35

1

5

18

- **Write a program to generate a random sequence of capital letters that does not include vowels**

```java
public class RandomChar {
    public static void main(String args[]){
        char character;
        int random_len = (int) (Math.random()*20);
        for(int i = 0; i < random_len; i++)
        {
            character = (char) ((Math.random()*26) + 65);
            if( character != 65 && character != 69 && character != 73
&& character != 79 && character != 85)
                System.out.print(character);
        }
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\RandomChar.java

QZFYCFPXFJYZZ

Class Work - 2

Q. Write "Help.java" Program.

```java
public class Help {
    public static void main(String[] args)
    throws java.io.IOException
    {
        char choice,ignore;

        for(;;)
        {
            do
            {
                System.out.println("Help on :");
                System.out.println(" 1. if");
                System.out.println(" 2. switch");
                System.out.println(" 3. for");
                System.out.println(" 4. while");
                System.out.println(" 5. do-while");
                System.out.println(" 6. break");
                System.out.println(" 7. continue\n");
                System.out.println("Choose one(q to quit) : ");

                choice = (char) System.in.read();
                do
                {
                    ignore = (char) System.in.read();
                }while(ignore!='\n');
            }while(choice < '1' | choice > '7' & choice != 'q');
            if(choice == 'q') break;

            System.out.println("\n");
```

```java
    switch(choice)
    {
        case '1':
            System.out.println("The if:\n");
            System.out.println("if(condition) statement;");
            System.out.println("else statement;");
            break;

        case '2':
            System.out.println("The switch:\n");
            System.out.println("switch(expression) {");
            System.out.println("    case constant:");
            System.out.println("        statement sequence");
            System.out.println("        break;");
            System.out.println("    //...");
            System.out.println("}");
            break;

        case '3':
            System.out.println("The for:\n");
            System.out.println("for(init; condition ; iteration)")
;
            System.out.println("   statement");
            break;

        case '4':
            System.out.println("The while\n");
            System.out.println("while(condition) statement");
            break;

        case '5':
            System.out.println("The do-while\n");
            System.out.println("do {");
            System.out.println("   statement;");
            System.out.println("} while(condition)");
            break;

        case '6':
```

```java
            System.out.println("The break\n");
            System.out.println("break; or break label;");
            break;

        case '7':
            System.out.println("The continue\n");
            System.out.println("continue; or continue label;");
            break;
        }
        System.out.println();
        }
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\Help.java

Help on :

1. if

2. switch

3. for

4. while

5. do-while

6. break

7. continue

Choose one(q to quit) :

1

The if:

if(condition) statement;

else statement;

Help on :

1. if

2. switch

3. for

4. while

5. do-while

6. break

7. continue

Choose one(q to quit) :

2

The switch:

switch(expression) {

```
  case constant:

    statement sequence

    break;

  //...

}
```

Help on :

1. if

2. switch

3. for

4. while

5. do-while

6. break

7. continue

Choose one(q to quit) :

q

Q. Rounding Errors.

```java
class RoundingError {
    public static void main(String args[]){
        double num, sroot, rerr;
        for(num = 1.0; num < 100.0; num++){
            sroot = Math.sqrt(num);
            System.out.println("Square root of " + num + " is " + sroot);

            rerr = num - (sroot * sroot);
            System.out.println("Rounding error is " + rerr + "\n");
        }
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\RoundingError.java

Square root of 1.0 is 1.0

Rounding error is 0.0

Square root of 2.0 is 1.4142135623730951

Rounding error is -4.440892098500626E-16

Square root of 3.0 is 1.7320508075688772

Rounding error is 4.440892098500626E-16

Square root of 4.0 is 2.0

Rounding error is 0.0

Square root of 5.0 is 2.23606797749979

Rounding error is -8.881784197001252E-16

Square root of 6.0 is 2.449489742783178

Rounding error is 8.881784197001252E-16

Square root of 7.0 is 2.6457513110645907

Rounding error is -8.881784197001252E-16

Square root of 8.0 is 2.8284271247461903

Rounding error is -1.7763568394002505E-15

Square root of 9.0 is 3.0

Rounding error is 0.0

Square root of 10.0 is 3.1622776601683795

Rounding error is -1.7763568394002505E-15

Square root of 11.0 is 3.3166247903554

Rounding error is 0.0

Square root of 12.0 is 3.4641016151377544

Rounding error is 1.7763568394002505E-15

Square root of 13.0 is 3.605551275463989

Rounding error is 1.7763568394002505E-15

Square root of 14.0 is 3.7416573867739413

Rounding error is 0.0

Square root of 15.0 is 3.872983346207417

Rounding error is -1.7763568394002505E-15

Square root of 16.0 is 4.0

Rounding error is 0.0

Square root of 17.0 is 4.123105625617661

Rounding error is 0.0

Square root of 18.0 is 4.242640687119285

Rounding error is 3.552713678800501E-15

Square root of 19.0 is 4.358898943540674

Rounding error is -3.552713678800501E-15

Square root of 20.0 is 4.47213595499958

Rounding error is -3.552713678800501E-15

Square root of 21.0 is 4.58257569495584

Rounding error is 0.0

Square root of 22.0 is 4.69041575982343

Rounding error is 0.0

Square root of 23.0 is 4.795831523312719

Rounding error is 3.552713678800501E-15

Square root of 24.0 is 4.898979485566356

Rounding error is 3.552713678800501E-15

Square root of 25.0 is 5.0

Rounding error is 0.0

Square root of 26.0 is 5.0990195135927845

Rounding error is 3.552713678800501E-15

Square root of 27.0 is 5.196152422706632

Rounding error is 0.0

Square root of 28.0 is 5.291502622129181

Rounding error is -3.552713678800501E-15

Square root of 29.0 is 5.385164807134504

Rounding error is 3.552713678800501E-15

Square root of 30.0 is 5.477225575051661

Rounding error is 0.0

Square root of 31.0 is 5.5677643628300215

Rounding error is 3.552713678800501E-15

Square root of 32.0 is 5.656854249492381

Rounding error is -7.105427357601002E-15

Square root of 33.0 is 5.744562646538029

Rounding error is 0.0

Square root of 34.0 is 5.830951894845301

Rounding error is 0.0

Square root of 35.0 is 5.916079783099616

Rounding error is 0.0

Square root of 36.0 is 6.0

Rounding error is 0.0

Square root of 37.0 is 6.082762530298219

Rounding error is 7.105427357601002E-15

Square root of 38.0 is 6.164414002968976

Rounding error is 7.105427357601002E-15

Square root of 39.0 is 6.244997998398398

Rounding error is 0.0

Square root of 40.0 is 6.324555320336759

Rounding error is -7.105427357601002E-15

Square root of 41.0 is 6.4031242374328485

Rounding error is 0.0

Square root of 42.0 is 6.48074069840786

Rounding error is 0.0

Square root of 43.0 is 6.557438524302

Rounding error is 7.105427357601002E-15

Square root of 44.0 is 6.6332495807108

Rounding error is 0.0

Square root of 45.0 is 6.708203932499369

Rounding error is -7.105427357601002E-15

Square root of 46.0 is 6.782329983125268

Rounding error is 0.0

Square root of 47.0 is 6.855654600401044

Rounding error is 0.0

Square root of 48.0 is 6.928203230275509

Rounding error is 7.105427357601002E-15

Square root of 49.0 is 7.0

Rounding error is 0.0

Square root of 50.0 is 7.0710678118654755

Rounding error is -7.105427357601002E-15

Square root of 51.0 is 7.14142842854285

Rounding error is -7.105427357601002E-15

Square root of 52.0 is 7.211102550927978

Rounding error is 7.105427357601002E-15

Square root of 53.0 is 7.280109889280518

Rounding error is 0.0

Square root of 54.0 is 7.3484692283495345

Rounding error is 0.0

Square root of 55.0 is 7.416198487095663

Rounding error is 0.0

Square root of 56.0 is 7.483314773547883

Rounding error is 0.0

Square root of 57.0 is 7.54983443527075

Rounding error is 0.0

Square root of 58.0 is 7.615773105863909

Rounding error is -7.105427357601002E-15

Square root of 59.0 is 7.681145747868608

Rounding error is 7.105427357601002E-15

Square root of 60.0 is 7.745966692414834

Rounding error is -7.105427357601002E-15

Square root of 61.0 is 7.810249675906654

Rounding error is 7.105427357601002E-15

Square root of 62.0 is 7.874007874011811

Rounding error is 0.0

Square root of 63.0 is 7.937253933193772

Rounding error is -7.105427357601002E-15

Square root of 64.0 is 8.0

Rounding error is 0.0

Square root of 65.0 is 8.06225774829855

Rounding error is 1.4210854715202004E-14

Square root of 66.0 is 8.12403840463596

Rounding error is -1.4210854715202004E-14

Square root of 67.0 is 8.18535277187245

Rounding error is 0.0

Square root of 68.0 is 8.246211251235321

Rounding error is 0.0

Square root of 69.0 is 8.306623862918075

Rounding error is 0.0

Square root of 70.0 is 8.366600265340756

Rounding error is 0.0

Square root of 71.0 is 8.426149773176359

Rounding error is 0.0

Square root of 72.0 is 8.48528137423857

Rounding error is 1.4210854715202004E-14

Square root of 73.0 is 8.54400374531753

Rounding error is 1.4210854715202004E-14

Square root of 74.0 is 8.602325267042627

Rounding error is 0.0

Square root of 75.0 is 8.660254037844387

Rounding error is -1.4210854715202004E-14

Square root of 76.0 is 8.717797887081348

Rounding error is -1.4210854715202004E-14

Square root of 77.0 is 8.774964387392123

Rounding error is -1.4210854715202004E-14

Square root of 78.0 is 8.831760866327848

Rounding error is -1.4210854715202004E-14

Square root of 79.0 is 8.888194417315589

Rounding error is 0.0

Square root of 80.0 is 8.94427190999916

Rounding error is -1.4210854715202004E-14

Square root of 81.0 is 9.0

Rounding error is 0.0

Square root of 82.0 is 9.055385138137417

Rounding error is -1.4210854715202004E-14

Square root of 83.0 is 9.1104335791443

Rounding error is 0.0

Square root of 84.0 is 9.16515138991168

Rounding error is 0.0

Square root of 85.0 is 9.219544457292887

Rounding error is 0.0

Square root of 86.0 is 9.273618495495704

Rounding error is 0.0

Square root of 87.0 is 9.327379053088816

Rounding error is -1.4210854715202004E-14

Square root of 88.0 is 9.38083151964686

Rounding error is 0.0

Square root of 89.0 is 9.433981132056603

Rounding error is 1.4210854715202004E-14

Square root of 90.0 is 9.486832980505138

Rounding error is 0.0

Square root of 91.0 is 9.539392014169456

Rounding error is 0.0

Square root of 92.0 is 9.591663046625438

Rounding error is 1.4210854715202004E-14

Square root of 93.0 is 9.643650760992955

Rounding error is 0.0

Square root of 94.0 is 9.695359714832659

Rounding error is -1.4210854715202004E-14

Square root of 95.0 is 9.746794344808963

Rounding error is 1.4210854715202004E-14

Square root of 96.0 is 9.797958971132712

Rounding error is 1.4210854715202004E-14

Square root of 97.0 is 9.848857801796104

Rounding error is 1.4210854715202004E-14

Square root of 98.0 is 9.899494936611665

Rounding error is 0.0

Square root of 99.0 is 9.9498743710662

Rounding error is 0.0

Q. Loop until S is typed.

```java
public class UntilSOccurs {
    public static void main(String args[])
    throws java.io.IOException{
        int i;
        char ignore;
        System.out.println("Press S to Stop.");
        for(i = 0; (char) System.in.read() != 'S'; i++)
        {
            if((char) System.in.read() != '\n')
            {
                System.out.println("Pass #" + i);
            }
            do{
                ignore = (char) System.in.read();
            }while(ignore != '\n');
        }
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\UntilSOccurs.java

Press S to Stop.

R

Pass #0

Q

Pass #3

S

Q. While demo.

```java
public class WhileDemo {
    public static void main(String args[])
    {
        char ch;
        ch = 'a';
        while(ch <= 'z')
        {
            System.out.println(ch);
            ch++;
        }
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\WhileDemo.java

a

b

c

d

e

f

g

h

i

j

k

l

m

n

o

p

q

r

s

t

u

v

w

x

y

z

Q. Do-While demo.

```java
public class DoWhileDemo {
    public static void main(String args[])
    throws java.io.IOException {
        char ch;
        char ignore;
        do{
                System.out.println("Press a key Followed by Enter : ")
;
                ch = (char) System.in.read();
                do{
                    ignore = (char) System.in.read();
                }while(ignore != '\n');
        }while(ch != 'q');
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\DoWhileDemo.java

Press a key Followed by Enter :

e

Press a key Followed by Enter :

w

Press a key Followed by Enter :

d

Press a key Followed by Enter :

Q

Q. Guess the letter game.

```java
public class Guessing {
    public static void main(String args[])
    throws java.io.IOException{

        char ch;
        char ignore;
        char answer = 'K';
        do{
            System.out.println("I'm thinking of a letter between A and
 Z.");
            System.out.println("Can you Guess it : ");

            ch = (char) System.in.read();

            do{
                ignore = (char) System.in.read();
            }while(ignore != '\n');

            if(ch == answer)
                System.out.println("**** Right ****");
            else{
                System.out.print("... Sorry, You're ");
                if(ch < answer) System.out.print("too low");
                else System.out.print("too high");
                System.out.print(" Try again! \n");
            }
        }while(answer != ch);
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\Guessing.java

I'm thinking of a letter between A and Z.

Can you Guess it :

t

... Sorry, You're too high Try again!

I'm thinking of a letter between A and Z.

Can you Guess it :

k

... Sorry, You're too high Try again!

I'm thinking of a letter between A and Z.

Can you Guess it :

K

**** Right ****

Q. Self test question.

```java
import java.util.Scanner;
public class SelfTest {
    public static void main(String[] args)
    {
        String input;
        char ch;
        Scanner scan = new Scanner(System.in);
        input = scan.nextLine();
        for(int i = 0; i < input.length(); i++)
        {
            ch = input.charAt(i);
            if(ch > 64 && ch < 91)
            {
                ch += 32;
                System.out.print(ch);
            }
            else{
                ch-=32;
                System.out.print(ch);
            }
        }
    }
}
```

Output :

PS E:\MCA\MCA SEM 3\JAVA> java .\SelfTest.java

rOLLwAla

RollWaLA

Q. Assertion – 1

```java
import java.util.Scanner;
public class Assertion1 {
    public static void main(String[] args)
    {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter your age : ");
        int value = scanner.nextInt();
        assert value >= 18 : "Eligible : ";
        System.out.println("Your entered age is " +  value);
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\Assertion1.java

Enter your age : 21

Your entered age is 21

Q. Assertion – 2

```java
public class Assertion2 {
    public static void main(String args[]) throws java.io.IOException
    {
        int value=10;

        assert value>=20 : "Eligible";
        System.out.println("Value : " + value);
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\Assertion2.java

Value : 10

Q. Example of Scanner

```java
import java.util.Scanner;
public class Scan {
    public static void main(String args[]) throws java.io.IOException {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter Your Name : ");
        String name = scanner.nextLine();
        System.out.print("Enter Your age : ");
        int age = scanner.nextInt();
        System.out.print("Enter Your salary : ");
        double salary = scanner.nextDouble();
        System.out.println("Name is:" + name);
        System.out.println("Age  is:" + age);
        System.out.println("Salary is:" + salary);

    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\Scan.java

Enter Your Name : Pradip

Enter Your age : 21

Enter Your salary : 25000

Name is:Pradip

Age is:21

Salary is:25000.0

Q. System Properties

```java
import java.util.*;
public class GetProperties{
    public static void main(String[] args)
    {
        Properties properties = System.getProperties();
        properties.list(System.out);

    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\GetProperties.java

-- listing properties --

java.specification.version=13

sun.cpu.isalist=amd64

sun.jnu.encoding=Cp1252

java.class.path=.

java.vm.vendor=Oracle Corporation

sun.arch.data.model=64

user.variant=

java.vendor.url=https://java.oracle.com/

java.vm.specification.version=13

os.name=Windows 10

sun.java.launcher=SUN_STANDARD

user.country=IN

sun.boot.library.path=C:\Program Files\Java\jdk-13.0.1\bin

sun.java.command=jdk.compiler/com.sun.tools.javac.laun...

jdk.debug=release

sun.cpu.endian=little

user.home=C:\Users\Xtrem

user.language=en

sun.stderr.encoding=cp850

java.specification.vendor=Oracle Corporation

java.version.date=2019-10-15

java.home=C:\Program Files\Java\jdk-13.0.1

file.separator=\

java.vm.compressedOopsMode=32-bit

line.separator=

sun.stdout.encoding=cp850

java.vm.specification.vendor=Oracle Corporation

java.specification.name=Java Platform API Specification

jdk.module.main.class=com.sun.tools.javac.launcher.Main

user.script=

sun.management.compiler=HotSpot 64-Bit Tiered Compilers

java.runtime.version=13.0.1+9

user.name=Xtrem

jdk.launcher.sourcefile=D:\MCA\MCA SEM 3\JAVA\Practice\.\GetP...

path.separator=;

os.version=10.0

java.runtime.name=Java(TM) SE Runtime Environment

file.encoding=Cp1252

java.vm.name=Java HotSpot(TM) 64-Bit Server VM

java.vendor.url.bug=https://bugreport.java.com/bugreport/

java.io.tmpdir=C:\Users\Xtrem\AppData\Local\Temp\

java.version=13.0.1

user.dir=D:\MCA\MCA SEM 3\JAVA\Practice

os.arch=amd64

java.vm.specification.name=Java Virtual Machine Specification

sun.os.patch.level=

java.library.path=C:\Program Files\Java\jdk-13.0.1\bin;...

java.vm.info=mixed mode, sharing

java.vendor=Oracle Corporation

java.vm.version=13.0.1+9

sun.io.unicode.encoding=UnicodeLittle

java.class.version=57.0

Q. 2D Array

```java
public class TwoDeminsionArray {
    public static void main(String args[]){
        int arr[][] = { {2,7,9},{3,6,1},{7,4,2} };
        for(int i = 0; i < 3; i++)
        {
            for(int j = 0; j < 3; j++)
            {
                System.out.print(arr[i][j] + " ");
            }
            System.out.println();
        }
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\TwoDeminsionArray.java

2 7 9

3 6 1

7 4 2

Q. 3D Array

```java
public class ThreeDeminsion {
    public static void main(String args[]){
        int arr[][][] = { { {2,7,9},{3,6,1},{7,4,2} }, { {3,6,1},{2,7,9},{7,4
,2} },{ {2,7,9},{7,4,2},{3,6,1} } };
        for(int i = 0; i < 3; i++)
        {
            for(int j = 0; j < 3; j++)
            {
                for(int k = 0; k < 3; k++)
                {
                    System.out.print(arr[i][j][k] + " ");
                }
                System.out.println();
            }
            System.out.println();
        }
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\ThreeDeminsion.java

2 7 9

3 6 1

7 4 2

3 6 1

2 7 9

7 4 2

2 7 9

7 4 2

3 6 1

Q. Jagged Array

```java
public class JaggedArray {
    public static void main(String[] args){
        int ja[][] = {{1,2,3},{1,2,3,4}};
        for(int i = 0; i < 2; i++)
        {
            for(int j = 0; j < 3 + i; j++)
            {
                System.out.print(ja[i][j] + " ");
            }
            System.out.println("\n");
        }
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\JaggedArray.java

1 2 3

1 2 3

Q. Array 1 Example

```java
import java.util.Arrays;
public class ArrayExample{
    public static void main(String[] args) {
        int arr[]=new int [5];

        for(int i=5;i>0;i--)
            arr[5-i] = i;

        Arrays.sort(arr);

        for(int i=0;i<5;++i)
            System.out.print(arr[i]);
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\ArrayExample.java

12345

Q. String Collection

```java
public class StringCollection {
    public static void main(String[] args){
        String Phrase = "The Quick Brown fox Jumped over the lazy dog";
        int vowels = 0;
        for(char ch : Phrase.toCharArray()){
            ch = Character.toLowerCase(ch);
            if(ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u')

                ++vowels;
        }
        System.out.println("The Phrase contains " + vowels + " Vowels");
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\StringCollection.java

The Phrase contains 12 Vowels

Q. Count Vowels

```java
public class StringCollection {
    public static void main(String[] args){
        String Phrase = "The Quick Brown fox Jumped over the lazy dog";
        int vowels = 0;
        for(char ch : Phrase.toCharArray()){
            ch = Character.toLowerCase(ch);
            if(ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u')

                ++vowels;
        }
        System.out.println("The Phrase contains " + vowels + "  Vowels");
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\StringCollection.java

The Phrase contains 12 Vowels

Q. Test String

```java
public class TestString {
    public static void main(String[] args){
        String s1 = "Test";
        String s2 = "Test";
        String s3 = new String("Test");
        final String s4 = s3.intern();
        System.out.println(s1 == s2);
        System.out.println(s2 == s3);
        System.out.println(s3 == s4);
        System.out.println(s1 == s3);
        System.out.println(s1 == s4);

        System.out.println(s1.equals(s2));
        System.out.println(s2.equals(s3));
        System.out.println(s3.equals(s4));
        System.out.println(s1.equals(s4));
        System.out.println(s1.equals(s3));
    }
}
```

Output:

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\TestString.java

true

false

false

false

true

true

true

true

true

true

Q. Lucky Star

```java
public class LuckyStar {
    public static void main(String[] args) {
        String[] stars = {
            "Robert Redford" , "Marilyn Monroe",
            "Boris Karloff" , "Lassie",
            "Hopalong Cassidy", "Trigger"
            };
        System.out.println("Your lucky star for today is " + stars[(int)(stars.length*Math.random())]);
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\LuckyStar.java

Your lucky star for today is Boris Karloff

Q. Join String

```java
public class JoinString {
    public static void main(String[] args) {
        String firstString = "Many ";
        String secondString = "hands ";
        String thirdString = "make light work";
        String myString;

        myString = firstString + secondString + thirdString;
        System.out.println(myString);

        int numHands = 99;
        myString = numHands + " " + secondString + thirdString;
        System.out.println(myString);


        myString = "fifty five is " + 5 + 5;
        System.out.println(myString);

        myString = 5 + 5 + " is ten ";
        System.out.println(myString);
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\JoinString.java

Many hands make light work

99 hands make light work

fifty five is 55

10 is ten

Q. Match String

```java
public class MatchStrings {
    public static void main(String[] args) {
        String string1 = "Too many ";
        String string2 = "cooks";
        String string3 = "Too many cooks";

        string1 += string2;

        System.out.println("Test 1");
        System.out.println("string3 is now: " + string3);
        System.out.println("string1 is now: " + string1);
        if(string1 == string3)
            System.out.println("string1 == string3 is true." +" string1 and strin
g3 point to the same string");
        else
            System.out.println("string1 == string3 is false." +" string1 and stri
ng3 do not point to the same string");

        string3 = string1;

        System.out.println("\n\nTest  2");
        System.out.println("string3 is now: " + string3);
        System.out.println("string1 is now: " + string1);
        if(string1 == string3)
            System.out.println("string1 == string3 is true." +" string1 and strin
g3 point to the same string");
        else
            System.out.println("string1 == string3 is false."+" string1 and strin
g3 do not point to the same string");
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\MatchStrings.java

Test 1

string3 is now: Too many cooks

string1 is now: Too many cooks

string1 == string3 is false. string1 and string3 do not point to the same string

Test 2

string3 is now: Too many cooks

string1 is now: Too many cooks

string1 == string3 is true. string1 and string3 point to the same string

Q. Match String (immutable string)

```java
class Immutablestr{
    public static void main(String args[]){
        String s = "MCA 3 ";
        s = s.concat("Placement");
        System.out.println(s);
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\Immutablestr.java

MCA 3 Placement

Q. All Method of String

```java
public class AllStringOperation {
    public static void main(String args[]){
        String str1 = "when it comes to web programming, java is #1.";
        String str2 = new String(str1);
        String str3 = "java strings are powerful.";
        int result,idx;
        char ch;

        System.out.println("length of str1 : "+str1.length());

        //display str1,one char at a time.
        for(int i=0; i<str1.length();i++)
            System.out.print(str1.charAt(i));

        System.out.println();

        if(str1.equals(str2))
            System.out.println("str1 equals str2");
        else
            System.out.println("str1 does not equal str2");

        if(str1.equals(str3))
            System.out.println("str1 equals str3");
        else
            System.out.println("str1 does not equal str3");

        result = str1.compareTo(str3);
        if(result == 0)
            System.out.println("str1 and str3 are equal");
        else if(result<0)
            System.out.println("str1 is less than str3");
        else
            System.out.println("str1 is greater than str3");

        idx = str2.indexOf("one");
        System.out.println("index of first occurence of one : "+idx);
        idx = str2.lastIndexOf("one");
        System.out.println("index of last occurence of one : "+idx);
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\AllStringOperation.java

length of str1 : 45

when it comes to web programming, java is #1.

str1 equals str2

str1 does not equal str3

str1 is greater than str3

index of first occurence of one : -1

index of last occurence of one : -1

Q. Array of String

```java
public class ArrayString {
    public static void main(String args[]){
        String strs [] = { "this","is","a","test."};
        System.out.print("original array : ");
        for(String s : strs)
            System.out.print(s+" ");

        System.out.println("\n");

        strs[1] = "was";
        strs[3] = "test , tool";
        System.out.print("Modified array : ");
        for(String s : strs)
            System.out.print(s+" ");
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\ArrayString.java

original array : this is a test.

Modified array : this was a test , tool

Q. Sub-String

```java
public class SubString {
    public static void main(String args[]){
        String orgstr = "java makes the web move.";

        String substr = orgstr.substring(5,18);

        System.out.println("orgstr : "+orgstr);
        System.out.println("substr : "+substr);
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\SubString.java

orgstr : java makes the web move.

substr : makes the web

Q. String Switch

```java
public class StringSwitch {
    public static void main(String args[]){
        String command = "connect";

        switch(command){
            case "connect" :
                System.out.println("connecting");
                break;
            case "cancel" :
                System.out.println("canceling");
                break;
            case "disconnect" :
                System.out.println("disconnecting");
                break;
            default :
                System.out.println("command error");
                break;
        }
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\StringSwitch.java

Connecting

Q. Using a StringBuffer Object to Assemble a String

```java
public class UseStringBuffer {
    public static void main(String[] args) {
        StringBuffer sentence = new StringBuffer(20);
        System.out.println("\nStringBuffer object capacity is "+ sentence.capacit
y()+" and string length is "+sentence.length());

        String[] words = {"Too" , "many", "cooks", "spoi", "the" , "broth"};
        sentence.append(words[0]);
        for(int i = 1 ; i<words.length ; i++)
            sentence.append(' ').append(words[i]);

        System.out.println("\nString in StringBuffer object is:\n" +sentence.toSt
ring());
        System.out.println("StringBuffer object capacity is now "+ sentence.capac
ity()+" and string length is "+sentence.length());

        sentence.insert(sentence.lastIndexOf("cooks")+4,"ie");
        sentence.insert(sentence.lastIndexOf("broth")+5, "er");
        System.out.println("\nString in StringBuffer object is:\n "+ sentence);
        System.out.println("StringBuffer object capacity is now "+ sentence.capac
ity()+" and string length is "+sentence.length());
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\UseStringBuffer.java

StringBuffer object capacity is 20 and string length is 0

String in StringBuffer object is:

Too many cooks spoi the broth

StringBuffer object capacity is now 42 and string length is 29

String in StringBuffer object is:

 Too many cookies spoi the brother

StringBuffer object capacity is now 42 and string length is 33

Q. Exciting Concordance Entries

```java
public class FindCharacter {
    public static void main(String[] args) {

        String text = "To be or not to be, that is the question;" + " Whether 'ti
s nobler in the mind to suffer" + " the slings and arrows of outrageous fortune,"
         + " or to take arms against a sea of troubles," + " and by opposing end
them?";
        int andCount = 0;
        int theCount = 0;
        int index = -1;
        String andStr = "and";
        String theStr = "the";

        index =text.indexOf(andStr);
        while(index >= 0) {
            ++andCount;
            index += andStr.length();
            index = text.indexOf(andStr, index);
        }

        index = text.lastIndexOf(theStr);
        while(index >= 0) {
            ++theCount;
            index -= theStr.length();
            index = text.lastIndexOf(theStr, index);
        }
        System.out.println("The text contains " + andCount + " ands\n" + "The text
 contains " + theCount + " thes");
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\FindCharacter.java

The text contains 2 ands

The text contains 5 thes

Q. String Tokenizing

```java
public class StringToken {
    public static void main(String[] args) {
        String text = "To be or not to be, that is the question.";
        String delimiters = "[, .]";
        int[] limits = {0, -1};

        for(int limit : limits) {
            System.out.println("\nAnalysis with limit = " + limit);
            String[] tokens = text.split(delimiters, limit);
            System.out.println("Number of tokens: " + tokens.length);
            for(String token : tokens) {
                System.out.println(token);
            }
        }
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\StringToken.java

Analysis with limit = 0

Number of tokens: 11

To

be

or

not

to

be

that

is

the

question

Analysis with limit = -1

Number of tokens: 12

To

be

or

not

to

be

that

is

the

question

Excercise - 2

Q. Exercise

- **Month And Average**

```java
public class MonthAndAverage {
    public static void main(String args[]) {
        String[] monthNames = {
                            "January"  , "February", "March"   , "April",
                            "May"      , "June"    , "July"    , "August",

                            "September","October"  , "November", "December
"
                          };

        double average = 0.0;
        double[] numbers = new double[12];

        for(int i = 0 ; i<numbers.length ; i++) {
          numbers[i] = Math.random()*100.0;
          System.out.println(monthNames[i] + " Generated  " + numbers[i]);
          average += numbers[i];
        }
        average /= numbers.length;

        System.out.println("\nAverage of Random Generated numbers is " + aver
age);
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice\23> java .\MonthAndAverage.java

January Generated 14.796351800091923

February Generated 39.711269832223785

March Generated 58.32386495973172

April Generated 19.083891467683557

May Generated 78.52264624324705

June Generated 72.46307769949362

July Generated 64.2760130349091

August Generated 51.97756817789709

September Generated 17.767136593640963

October Generated 71.2641366807655

November Generated 82.89356693449919

December Generated 49.55683721304006

Average of Random Generated numbers is 51.71969671976862

- **Tables**

```java
public class Tables {
    final static int TABLESIZE = 12;
    public static void main(String[]args) {
        int[][] table = new int[TABLESIZE][TABLESIZE];

        for(int i = 0 ; i<table.length ; i++) {
            for(int j = 0 ; j<table[i].length ; j++)
                table[i][j] = (i+1)*(j+1);
        }

        System.out.print("        ");

        for(int j = 1 ; j<=table[0].length ; j++) {
            System.out.print((j<10 ? "    ": "   ") + j);
        }

        System.out.println("\n_____--");

        for(int i=0; i<table.length; i++) {
            System.out.print("Row" + (i<9 ? "   ":" ") + (i+1) +":");

            for(int j=0; j<table[i].length; j++) {
```

```
            System.out.print((table[i][j]<10 ? "   " : table[i][j]<100 ? "  " :
" ") + table[i][j]);
        }

        System.out.println();
    }
  }
 }
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice\23> java .\Tables.java

        1 2 3 4 5 6 7 8 9 10 11 12

Row 1: 1   2   3   4   5 6 7 8 9 10 11 12

Row 2: 2   4   6   8   10 12 14 16 18 20 22 24

Row 3: 3   6   9   12   15 18 21 24 27 30 33 36

Row 4: 4   8 12   16 20 24 28 32 36 40 44 48

Row 5: 5 10 15 20 25 30 35 40 45 50 55 60

Row 6: 6 12 18 24 30 36 42 48 54 60 66 72

Row 7: 7 14 21 28 35 42 49 56 63 70 77 84

Row 8: 8 16 24 32 40 48 56 64 72 80 88 96

Row 9: 9 18 27 36 45 54 63 72 81 90 99 108

Row 10: 10 20 30 40 50 60 70 80 90 100 110 120

Row 11: 11 22 33 44 55 66 77 88 99 110 121 132

Row 12: 12 24 36 48 60 72 84 96 108 120 132 144

- **Sort**

```java
public class Sort {
    public static void main(String args[]) {

        String text = "Lorem Ipsum is simply dummy text of the printing and typeset
ting industry " +
                      "It is a long established fact that a reader will be distract
ed by the readable content of a page when looking at its layout ";

        int count = 0;
        boolean isWord = false;
        for (int i = 0 ; i<text.length() ; i++) {
            if(isWord) {
                if(Character.isLetter(text.charAt(i)) || text.charAt(i) == '\'')
                    continue;
                else
                    isWord = false;
            }
            else if(Character.isLetter(text.charAt(i))) {

                count++;
                isWord = true;

            }
        }


        String[] words = new String[count];


        int start = 0;
        int wordIndex = 0;

        isWord = false;
        for (int i = 0 ; i<text.length() ; i++) {
            if(!isWord) {
                if(Character.isLetter(text.charAt(i))) {
                    start = i;
                    isWord = true;
                }
            }
            else {
                if(Character.isLetter(text.charAt(i)) || text.charAt(i) == '\'')
                    continue;
```

```java
            else {
                isWord = false;
                words[wordIndex++] = text.substring(start,i);
            }
        }
    }

    if(wordIndex < words.length)
        words[wordIndex] = text.substring(start);


    String temp = null;
    boolean exchange = true;
    while(exchange) {
        exchange = false;
        for(int i = 1 ; i<words.length ; i++) {
            if(words[i-1].compareTo(words[i])>0) {
                temp = words[i];
                words[i] = words[i-1];
                words[i-1] = temp;
                exchange = true;
            }
        }
    }

    for(String word : words) {
        System.out.println(word);
    }

  }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice\23> java .\Sort.java

Ipsum

It

Lorem

a

a

a

and

at

be

by

content

distracted

dummy

established

fact

industry

is

is

its

layout

long

looking

of

of

page

printing

readable

reader

simply

text

that

the

the

typesetting

when

will

- **Dates**

```java
public class Date {
    public static void main(String args[]) {
        String[] dates = new String[10];
        String[] monthNames = {
                "January", "February", "March", "April",
                "May", "June", "July", "August", "September",
                "October", "November", "December"
                };

        int[] daysInMonth = {31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31};
        String ending = "th";
        String dayStr = null;
        String monthStr = null;
        String yearStr = null;
        char separator = '/';
        int day = 0;
        int month = 0;
```

```java
        int year = 0;
        int daysIncrement = 0;

        for(int i=0 ; i<dates.length ; i++) {
            year = (int)(100.0*Math.random());
            month = (int)(12.0*Math.random()) + 1;

            daysIncrement = (((year % 4 == 0) && !(year % 100 == 0)) || (year % 4
00 == 0)) && (month == 2) ? 1 : 0 ;

            day = (int)(Math.random()*(daysInMonth[month-1]+daysIncrement))+1;
            dates[i] = "" + day + separator + month + separator + (year<10 ? "0"
:"") + year;
            System.out.println(dates[i]);
        }

        int start = 0;
        int end = 0;

        System.out.println();
        for(String date : dates){
            start = 0;
            end = date.indexOf(separator,start);
            dayStr = date.substring(start, end);
            start = end+1;
            end = date.indexOf(separator,start);
            monthStr = date.substring(start, end);
            start = end+1;
            yearStr = date.substring(start);

            if(dayStr.length() == 1)
                switch(dayStr.charAt(0)) {
                    case '1':
                    ending = "st";
                    break;
                    case '2':
                    ending = "nd";
                    break;
                    case '3':
                    ending = "rd";
                    break;
                    default:
                    ending = "th";
                }
```

```java
            else if(dayStr.charAt(0) == '1')
                ending = "th";
            else
                switch(dayStr.charAt(1)) {
                    case '1':
                    ending = "st";
                    break;
                    case '2':
                    ending = "nd";
                    break;
                    case '3':
                    ending = "rd";
                    break;
                    default:
                    ending = "th";
                }


        System.out.println(dayStr + ending + " " +
            monthNames[monthStr.length() == 1 ? monthStr.charAt(0) -
 '1': 9 + monthStr.charAt(1) - '0'] + " " + "19" + yearStr);
        }
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice\23> java .\Date.java

14/11/55

9/2/85

6/1/93

28/12/07

21/2/78

12/2/04

3/11/13

31/1/31

1/6/85

6/1/61

14th November 1955

9th February 1985

6th January 1993

28th December 1907

21st February 1978

12th February 1904

3rd November 1913

31st January 1931

1st June 1985

6th January 1961

- **Reverse**

```java
public class Reverse {

    public static void main(String args[]) {
        String text = "Lorem Ipsum is simply dummy text of the printing and typesetting industry " +
        "\nIt is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout ";

        boolean isWord = false;
        int start = 0;
        StringBuffer reversedText = new StringBuffer();
```

```java
        StringBuffer word = new StringBuffer();
        for(int i = 0 ; i<text.length() ; i++) {
          if(!isWord) {
            if(Character.isLetter(text.charAt(i)))  {
              word.append(text.charAt(i));
              isWord = true;
            } else {
              reversedText.append(text.charAt(i));
            }
          } else {
            if(Character.isLetter(text.charAt(i)) || text.charAt(i) == '\'') {
              word.append(text.charAt(i));
              continue;
            }
            else {
              reversedText.append(word.reverse());
              reversedText.append(text.charAt(i));
              word.delete(0,word.length());
              isWord = false;
            }
          }
        }
        if(word.length()>0)
          reversedText.append(word.reverse());

        System.out.println(reversedText);
    }

}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice\23> java .\Reverse.java

meroL muspI si ylpmis ymmud txet fo eht gnitnirp dna gnittesepyt yrtsudni

tI si a gnol dehsilbatse tcaf taht a redaer lliw eb detcartsid yb eht elbadaer
tnetnoc fo a egap nehw gnikool ta sti tuoyal

Class Work - 3

Q. Reference Assignment

```java
public class AssignRef {
    public static void main(String args[]){
        int i;
        int num1[] = new int[10];
        int num2[] = new  int[10];

        for(i=0;i<10;i++)
            num1[i] = i;

        for(i=0;i<10;i++)
            num2[i] = -i;

        System.out.println("here is num1 : ");

        for(i=0;i<10;i++)
            System.out.print(num1[i]+" ");

        System.out.println();
        System.out.println("here is num2 :  ");

        for(i=0;i<10;i++)
            System.out.print(num2[i]+"  ");

        System.out.println();

        num2 = num1;

        System.out.println("here is num2 after assignment : ");

        for(i=0;i<10;i++)
            System.out.print(num2[i]+"  ");

        System.out.println();

        num2[3] = 99;

        System.out.println("here is num1 after change through num1 :  ");

        for(i=0;i<10;i++)
            System.out.print(num1[i]+"  ");

        System.out.println();
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\AssignRef.java

here is num1 :

0 1 2 3 4 5 6 7 8 9

here is num2 :

0 -1 -2 -3 -4 -5 -6 -7 -8 -9

here is num2 after assignment :

0 1 2 3 4 5 6 7 8 9

here is num1 after change through num1 :

0 1 2 99 4 5 6 7 8 9

Q. Queue Demo

```java
public class QueueDemo {
    public static void main(String args[]){
        Queue bigQ = new Queue(100);
        Queue smallQ = new Queue(4);
        char ch;

        System.out.println("Using bigQ to stire the alphabet");

        for (int i = 0; i < 26; i++)
            bigQ.put((char) ('A' + i));

        System.out.println("Contents of bigQ: ");
        for (int i = 0; i < 26; i++) {
            ch = bigQ.get();
            if(ch != (char) 0)
                System.out.println(ch);
        }
        System.out.println("\n");

        System.out.println("Using smallQ to generate errors");
        for (int i = 0; i < 5; i++) {
            System.out.println("Attempting to store " + (char) ('Z' - i));
            smallQ.put((char) ('Z' - i));
            System.out.println();
        }
        System.out.println();

        System.out.println("Contents of smallQ : ");
        for (int i = 0; i < 5; i++) {
            ch = smallQ.get();
            if(ch != (char) 0)
                System.out.println(ch);
        }
    }
}

class Queue {
    char q[];
    int putloc;
    int getloc;

    Queue(int size) {
        q = new char[size];
        putloc = getloc = 0;
    }

    void put(char ch) {
        if(putloc == q.length) {
```

```java
            System.out.println(" - Queue is Full.");
            return;
        }

        q[putloc++] = ch;
    }

    char get() {
        if(getloc == putloc) {
            System.out.println(" - Queue is Empty.");
            return (char) 0;
        }

        return q[getloc++];
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\QueueDemo.java

Using bigQ to stire the alphabet

Contents of bigQ:

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

Y

Z

Using smallQ to generate errors

Attempting to store Z

Attempting to store Y

Attempting to store X

Attempting to store W

Attempting to store V
 - Queue is Full.

Contents of smallQ :

Z

Y

X

W
 - Queue is Empty.

Q. TryGeometry

```java
import static java.lang.Math.sqrt;

public class TryGeometry {
    public static void main(String[] args) {
        Point start = new Point(0.0, 1.0);
        Point end = new Point(5.0, 6.0);
        System.out.println("Points created are " + start + " and " +  end);

        Line line1 = new Line(start, end);
        Line line2 = new Line(0.0, 3.0, 3.0, 0.0);
        System.out.println("Lines created are " + line1 + " and " + line2);

        System.out.println("Intersection is " + line2.intersects(line1));

        end.move(1.0, -5.0);
        System.out.println("Intersection is " +  line1.intersects(line2));

    }
}

class Point {
    // Coordinates of the point
    double x;
    double y;

    Point(double xVal, double yVal) {
        x = xVal;
        y = yVal;
    }

    Point(final Point oldPoint) {
        x = oldPoint.x;
        y = oldPoint.y;
    }

    void move(double xDelta, double yDelta) {

        x += xDelta;
        y += yDelta;
    }

    double distance(final Point aPoint) {
        return sqrt((x - aPoint.x)*(x - aPoint.x) + (y - aPoint.y)*(y -
 aPoint.y));
    }


    public String toString() {
```

```java
        return Double.toString(x) + ", " + y;
    }
}

class Line {
    Point start;
    Point end;

    Line(final Point start, final Point end) {
        this.start = new Point(start);
        this.end = new Point(end);
    }

    Line(double xStart, double yStart, double xEnd, double yEnd) {
        start = new Point(xStart, yStart);
        end = new Point(xEnd, yEnd);
    }

    double length() {
        return start.distance(end);
    }

    public String toString() {
        return "(" + start+ "):(" + end + ")";
    }

    Point intersects(final Line line1) {
        Point localPoint = new Point(0, 0);
        double num = (this.end.y - this.start.y)*(this.start.x -
 line1.start.x) -
        (this.end.x - this.start.x)*(this.start.y - line1.start.y);
        double denom = (this.end.y - this.start.y)*(line1.end.x  -
 line1.start.x) -
        (this.end.x - this.start.x)*(line1.end.y - line1.start.y);
        localPoint.x = line1.start.x + (line1.end.x -
 line1.start.x)*num/denom;
        localPoint.y = line1.start.y + (line1.end.y -
 line1.start.y)*num/denom;
        return localPoint;
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice\trygeometry> java .\TryGeometry.java

Points created are 0.0, 1.0 and 5.0, 6.0

Lines created are (0.0, 1.0):(5.0, 6.0) and (0.0, 3.0):(3.0, 0.0)

Intersection is 1.0, 2.0

Intersection is 1.0, 2.0

Q. Rabbit Out of Hats

```java
import java.util.Random;
public class MagicHat {
    static int maxRabbits = 5;
    static Random select = new Random();

    public MagicHat(String hatName) {
        this.hatName = hatName;
        rabbits = new Rabbit[1+select.nextInt(maxRabbits)];
        for(int i = 0; i < rabbits.length; i++) {
        rabbits[i] = new Rabbit();
        }
    }

    public String toString() {

        String hatString = "\n" + hatName + " contains:\n";
        for(Rabbit rabbit : rabbits) {
            hatString += " " + rabbit;
        }
    return hatString;
    }
    private String hatName;
    private Rabbit rabbits[];

    static class Rabbit {

    }

    static public void main(String[] args) {
        System.out.println(new MagicHat("Gray Topper"));
        System.out.println(new MagicHat("Black Topper"));
        System.out.println(new MagicHat("Baseball Cap"));
        MagicHat oldHat = new MagicHat("Old hat");
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\MagicHat.java

Gray Topper contains:
 MagicHat$Rabbit@275710fc MagicHat$Rabbit@525f1e4e
MagicHat$Rabbit@75f9eccc MagicHat$Rabbit@52aa2946

Black Topper contains:

 MagicHat$Rabbit@4de5031f MagicHat$Rabbit@67e2d983
MagicHat$Rabbit@5d47c63f

Baseball Cap contains:

 MagicHat$Rabbit@5ea434c8 MagicHat$Rabbit@3bbc39f8
MagicHat$Rabbit@4ae3c1cd   MagicHat$Rabbit@29f69090
MagicHat$Rabbit@568bf312

Q. Nested Class Demo

```java
public class NestedClassDemo {
    public static void main(String args[]){
        int x[] = {3,2,1,5,6,9,7,8};
        Outer outob = new Outer(x);
        outob.analyze();

    }
}
class Outer{
    int nums[];
    Outer(int n[]){
        nums=n;
    }
    void analyze(){
        Inner inob = new Inner();
        System.out.println("Minimum : "+inob.min());
        System.out.println("Maximum: "+inob.max());
        System.out.println("Average : "+inob.avg());
    }
    class Inner{
        int min(){
            int m = nums[0];
            for(int i =1; i<nums.length; i++)
                if(nums[i] < m) m = nums[i];
                    return m;
        }
        int max(){
            int m = nums[0];
            for(int i =1; i<nums.length; i++)
                if(nums[i] > m) m = nums[i];
                    return m;
        }
        int avg(){
            int a = 0;
            for(int i =1; i<nums.length; i++)
              a += nums[i];

               return a/nums.length;
        }
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\NestedClassDemo.java

Minimum : 1

Maximum: 9

Average : 4

Q. Interface

➔ Conversion.java

```java
package conversions;
public interface Conversion {
    double inchesToMillimeters (double inches);
    double ouncesToGrams(double ounces);
    double poundsToGrams(double pounds);
    double hpToWatts(double hp);
    double wattsToHP(double watts);
}
```

➔ ConversionFactors.java

```java
package conversions;
public interface ConversionFactors {
    double INCH_TO_MM = 25.4;
    double OUNCE_TO_GRAM = 28.349523125;
    double POUND_TO_GRAM = 453.5924;
    double HP_TO_WATT = 745.7;
    double WATT_TO_HP = 1.0/HP_TO_WATT;
}
```

➔ TryConversion.java

```java
package conversions;
import static conversions.ConversionFactors.*; // Import static members
public class TryConversion implements Conversion {
    public double wattsToHP (double watts) {
            return watts*WATT_TO_HP;
      }
    public double hpToWatts (double hp) {
            return hp*HP_TO_WATT;
       }
    public double ouncesToGrams(double ounces) {
             return ounces*OUNCE_TO_GRAM;
       }
    public double poundsToGrams(double pounds) {
            return pounds*POUND_TO_GRAM;
       }
    public double inchesToMillimeters(double inches) {
           return inches*INCH_TO_MM;
```

```
        }
    public static void main(String args[]) {
        int myWeightInPounds = 180;
        int myHeightInInches = 75;
        TryConversion converter = new TryConversion();
        System.out.println("My weight in pounds: " +myWeightInPounds +"  \t-
in grams: "+ (int)converter.poundsToGrams(myWeightInPounds));
        System.out.println("My height in inches: " + myHeightInInches+ "  \t
-in millimeters: "+ (int)converter.inchesToMillimeters(myHeightInInches));
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> javac -d . .\Conversion.java

PS D:\MCA\MCA SEM 3\JAVA\Practice> javac -d . .\ConversionFactors.java

PS D:\MCA\MCA SEM 3\JAVA\Practice> javac -d . .\TryConversion.java

PS D:\MCA\MCA SEM 3\JAVA\Practice> java conversions.TryConversion

My weight in pounds: 180   -in grams: 81646

My height in inches: 75   -in millimeters: 1905

Q. Defining Interface

➜ RemoteControl.java

```java
package remote;
public interface RemoteControl {
      boolean powerOnOff(); // Returns new state, on = true
      int volumeUp(int increment); // Returns new volume level
      int volumeDown(int decrement); // Returns new volume level
      void mute(); // Mutes sound output
      int setChannel(int channel); // Set the channel number and return it
      int channelUp(); // Returns new channel number
      int channelDown(); // Returns new channel number
}
```

➜ TV.java

```java
package remote;

import static java.lang.Math.max;
import static java.lang.Math.min;

public class TV implements RemoteControl {

    public TV(String make, int screensize) {
        this.make = make;
        this.screensize = screensize;
    }

    public boolean powerOnOff() {
        power = !power;
        System.out.println(make + " "+ screensize + " inch TV power "+ (power ? "
on.":"off."));
        return power;
    }

    public int volumeUp(int increment) {
        if(!power)
            return 0;

        volume += increment;
        volume = min(volume, MAX_VOLUME);
```

```java
        System.out.println(make + " "+ screensize + " inch TV volume level: "+ vo
lume);
        return volume;
    }

    public int volumeDown(int decrement) {
        if(!power)
            return 0;

        volume -= decrement;
        volume = max(volume, MIN_VOLUME);
        System.out.println(make + " "+ screensize + " inch TV volume level: " + v
olume);
        return volume;
    }
    public void mute(){
        if(!power)
            return;

        volume = MIN_VOLUME;
        System.out.println(make + " "+ screensize + " inch TV volume level: "+ vo
lume);
    }

    public int setChannel(int newChannel) {
        if(!power)
            return 0;

        if(newChannel>=MIN_CHANNEL && newChannel<=MAX_CHANNEL)
            channel = newChannel;

        System.out.println(make + " "+ screensize + " inch TV tuned to channel: "
+ channel);
        return channel;
    }

    public int channelUp(){
        if(!power)
            return 0;

        channel = channel<MAX_CHANNEL ? ++channel : MIN_CHANNEL;
        System.out.println(make + " "+ screensize + " inch TV tuned to channel: "
+ channel);
        return channel;
```

```
        }

    public int channelDown(){
        if(!power)
            return 0;

        channel = channel>MIN_CHANNEL ? --channel : MAX_CHANNEL;
        System.out.println(make + " "+ screensize + " inch TV tuned to channel: "
+ channel);
        return channel;
    }
    private String make = null;
    private int screensize = 0;
    private boolean power = false;
    private int MIN_VOLUME = 0;
    private int MAX_VOLUME = 100;
    private int volume = MIN_VOLUME;
    private int MIN_CHANNEL = 0;
    private int MAX_CHANNEL = 999;
    private int channel = 0;
}
```

➔ **VCR.java**

```
package remote;
import static java.lang.Math.max;
import static java.lang.Math.min;
public class VCR implements RemoteControl {
    public VCR(String make) {
        this.make = make;
    }

    public boolean powerOnOff() {
        power = !power;
        System.out.println(make + " VCR power "+ (power ? "on.":"off."));
        return power;
    }

    public int volumeUp(int increment) {
        if(!power)
            return 0;
```

```java
        volume += increment;
        volume = min(volume, MAX_VOLUME);
        System.out.println(make + " VCR volume level: "+ volume);
        return volume;
    }

    public int volumeDown(int decrement) {
        if(!power)
            return 0;

        volume -= decrement;
        volume = max(volume, MIN_VOLUME);
        System.out.println(make + " VCR volume level: "+ volume);
        return volume;
    }

    public void mute(){
        if(!power)
            return;

        volume = MIN_VOLUME;
        System.out.println(make + " VCR volume level: "+volume);
    }
    public int setChannel(int newChannel) {
        if(!power)
            return 0;

        if(newChannel>=MIN_CHANNEL && newChannel<=MAX_CHANNEL)
            channel = newChannel;

        System.out.println(make + " VCR tuned to channel: "+ channel);
        return channel;
    }

    public int channelUp(){
        if(!power)
            return 0;

        channel = channel<MAX_CHANNEL ? ++channel : MIN_CHANNEL;
        System.out.println(make + " VCR tuned to channel: "+ channel);
        return channel;
    }
```

```java
    public int channelDown(){
        if(!power)
            return 0;

        channel = channel>MIN_CHANNEL ? --channel : MAX_CHANNEL;
        System.out.println(make + " VCR tuned to channel: "+ channel);
        return channel;
    }
    private String make = null;
    private boolean power = false;
    private int MIN_VOLUME = 0;
    private int MAX_VOLUME = 100;
    private int volume = MIN_VOLUME;
    private int MIN_CHANNEL = 0;
    private int MAX_CHANNEL = 99;
    private int channel = 0;
}
```

➔ **TryRemoteControl.java**

```java
package remote;
import static java.lang.Math.random;
public class TryRemoteControl {
    public static void main(String args[]) {
        RemoteControl remote = null;
        for(int i = 0 ; i<5 ; i++) {
            if(random()<0.5)
                remote = new TV(random()<0.5 ? "Sony" : "Hitachi",random()<0.5 ?
32 : 28);
            else
                remote = new VCR(random()<0.5 ? "Panasonic": "JVC");
        remote.powerOnOff();
        remote.channelUp();
        remote.volumeUp(10);
        }
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> javac -d . .\RemoteControl.java

PS D:\MCA\MCA SEM 3\JAVA\Practice> javac -d . .\TV.java

PS D:\MCA\MCA SEM 3\JAVA\Practice> javac -d . .\VCR.java

PS D:\MCA\MCA SEM 3\JAVA\Practice> javac -d . .\TryRemoteControl.java

PS D:\MCA\MCA SEM 3\JAVA\Practice> java remote.TryRemoteControl

Panasonic VCR power on.

Panasonic VCR tuned to channel: 1

Panasonic VCR volume level: 10

JVC VCR power on.

JVC VCR tuned to channel: 1

JVC VCR volume level: 10

Hitachi 32 inch TV power on.

Hitachi 32 inch TV tuned to channel: 1

Hitachi 32 inch TV volume level: 10

JVC VCR power on.

JVC VCR tuned to channel: 1

JVC VCR volume level: 10

Hitachi 32 inch TV power on.

Hitachi 32 inch TV tuned to channel: 1

Hitachi 32 inch TV volume level: 10

Q. Queue Interface

➜ ICharQ.java

```java
public interface ICharQ
{
    //Put a character into the queue
    void put(char ch);
    //Get a character from the queue
    char get();
}
```

➜ IQDemo.java

```java
class FixedQueue implements ICharQ{
    private char q[];
    private int putloc,getloc;

    public FixedQueue(int size)
    {
        q = new char[size];
        putloc = getloc = 0;
    }


    public void put(char ch)
    {
        if(putloc == q.length)
        {
            System.out.println("Alert : Queue is full");
            return;
        }
        q[putloc++] = ch;
    }


    public char get()
    {
        if(getloc == putloc)
        {
            System.out.println("Alert : Queue is empty");
            return (char) 0;
        }
```

```java
        return q[getloc++];
    }
}

class CircularQueue implements ICharQ
{
    private char q[];
    private int putloc,getloc;

    public CircularQueue(int size)
    {
        q = new char[size+1];
        putloc = getloc = 0;
    }

    public void put(char ch)
    {
        if(putloc+1 == getloc | ((putloc == q.length-1) & (getloc == 0)))
        {
            System.out.println("Alert : Queue is full");
            return;
        }
        q[putloc++] = ch;
        if(putloc == q.length)
        {
            putloc = 0;
        }
    }

    public char get()
    {
        if(getloc == putloc)
        {
            System.out.println("Alert : Queue is empty");
            return (char) 0;
        }
        char ch = q[getloc++];
        if(getloc == q.length) getloc = 0;
        return ch;
    }
}

class DynQueue implements ICharQ
{
```

```java
    private char q[];
    private int putloc,getloc;

    public DynQueue(int size)
    {
        q = new char[size];
        putloc = getloc = 0;
    }

    public void put(char ch)
    {
        if(putloc == q.length)
        {

            char t[] = new char[q.length * 2];

            for(int i=0; i < q.length; i++)
                t[i] = q[i];
            q = t;
        }
        q[putloc++] = ch;
    }

    public char get()
    {
        if(getloc == putloc)
        {
            System.out.println("Alert : Queue is empty");
            return (char) 0;
        }
        return q[getloc++];
    }
}

class IQDemo
{
    public static void main(String[] args)
    {
        FixedQueue fixedQueue = new FixedQueue(10);
        DynQueue dynQueue = new DynQueue(5);
        CircularQueue circularQueue = new CircularQueue(10);

        ICharQ iQ;
        char ch;
```

```java
        int i;
        iQ = fixedQueue;

        for(i=0; i < 10;i++)
        {
            iQ.put((char) ('A' + i));
        }


        System.out.println("Contents of FixedQueue: ");
        for(i = 0; i < 10; i++)
        {
            ch = iQ.get();
            System.out.print(ch);
        }
        System.out.println();

        iQ = dynQueue;

        for(i=0; i < 10;i++)
        {
            iQ.put((char) ('Z' - i));
        }


        System.out.println("Contents of DynQueue: ");
        for(i = 0; i < 10; i++)
        {
            ch = iQ.get();
            System.out.print(ch);
        }
        System.out.println();

        iQ = circularQueue;

        for(i=0; i < 10;i++)
        {
            iQ.put((char) ('A' + i));
        }


        System.out.println("Contents of CircularQueue: ");
        for(i = 0; i < 10; i++)
        {
```

```java
            ch = iQ.get();
            System.out.print(ch);
        }
        System.out.println();


        for(i=10; i < 20;i++)
        {
            iQ.put((char) ('A' + i));
        }


        System.out.println("Contents of CircularQueue: ");
        for(i = 0; i < 10; i++)
        {
            ch = iQ.get();
            System.out.print(ch);
        }
        System.out.println("\nStore and consume from circular queue");


        for(i = 0; i < 20; i++)
        {
            iQ.put((char) ('A' +i));
            ch = iQ.get();
            System.out.print(ch);
        }
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> javac .\ICharQ.java

PS D:\MCA\MCA SEM 3\JAVA\Practice> javac .\IQDemo.java

PS D:\MCA\MCA SEM 3\JAVA\Practice> java IQDemo

Contents of FixedQueue:

ABCDEFGHIJ

Contents of DynQueue:

ZYXWVUTSRQ

Contents of CircularQueue:

ABCDEFGHIJ

Contents of CircularQueue:

KLMNOPQRST

Store and consume from circular queue

ABCDEFGHIJKLMNOPQRST

Q. Generic Functional Interface

```java
class GenericFunctionalInterface
{
    public static void main(String[] args)
    {
        SomeTest<Integer> isFactor = (n,d) -> (n%d) == 0;

        if(isFactor.test(10,2))
            System.out.println("2 is factor of 10");
        System.out.println();

        SomeTest<Double> isFactorD = (n,d) -> (n%d) == 0;

        if(isFactorD.test(212.0,4.0))
            System.out.println("4 is factor of 212");

        System.out.println();

        SomeTest<String> isIn = (a, b) -> a.indexOf(b) != -1;
        String str = "Generic Functional Interface";
        if(isIn.test(str, "face"))
            System.out.println("face is found");
        else
            System.out.println("face is no found");
    }
}
interface SomeTest<T>
{
    boolean test(T n, T m);
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\GenericFunctionalInterface.java

2 is factor of 10

4 is factor of 212

face is found

Q. Lambda Argument Demo

```java
public class LambdaArgumentDemo {
    static String changeStr(StringFunc sf, String s) {
        return sf.func(s);
    }

    public static void main(String args[]) {
        String inStr = "Lambda Expression Expand Java";
        String outStr;

        System.out.println("Here is input string : " + inStr);

        StringFunc reverse = (str) -> {
            String result = "";
            for (int i = str.length()-1; i >= 0; i--)
                result += str.charAt(i);

            return result;
        };

        outStr = changeStr(reverse, inStr);
        System.out.println("The string reversed : " +  outStr);

        outStr = changeStr((str) -> str.replace(' ', '-'), inStr);
        System.out.println("The string with spaces replaced : " + outStr);

        outStr = changeStr((str) -> {
                String result = "";
                char ch;

                for (int i = 0; i < str.length(); i++) {
                    ch = str.charAt(i);
                    if(Character.isUpperCase(ch))
                        result += Character.toLowerCase(ch);
                    else
                        result += Character.toUpperCase(ch);
                }
                return result;
        }, inStr);
        System.out.println("The string in reversed case : " +  outStr);
    }


}
```

```
interface StringFunc {
    String func(String str);
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\LambdaArgumentDemo.java

Here is input string : Lambda Expression Expand Java

The string reversed : avaJ dnapxE noisserpxE adbmaL

The string with spaces replaced : Lambda-Expression-Expand-Java

The string in reversed case : lAMBDA eXPRESSION eXPAND jAVA

Q. Lambda Exception Demo

```java
import java.io.*;

  class LambdaExceptionDemo{
    public static void main(String args[])
    {
        double[] values = {1.0,2.0,3.0,4.0};
        MyIOAction myIO = (rdr) -> {
            int ch = rdr.read();
            return true;
        };
    }
}
interface MyIOAction{
    boolean ioAction(Reader rdr) throws IOException;
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\LambdaExceptionDemo.java

PS D:\MCA\MCA SEM 3\JAVA\Practice>

Q. Package ExtInterface

➔ ExtInterface.java

```java
package extInterface;
public interface ExtInterface {
    public void method1();
    public void method2();

}
```

➔ TestExtInterface.java

```java
package extInterface;

import java.util.Scanner;

class TestExtInterface implements ExtInterface{
    public void method1() {
        System.out.println("Implementation of Method1");
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter number to find the Square root in java:");
        double square = sc.nextDouble();
        double squareRoot = Math.sqrt(square);
        System.out.printf("Square root of number: %f is -
> %f",square,squareRoot);

    }
    public void method2() {
        System.out.println("Implementation of method2");
    }
    public static void main(String[] args) {
        ExtInterface obj  = new TestExtInterface();
        obj.method1();
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice\35> javac –d . .\ExtInterface.java

PS D:\MCA\MCA SEM 3\JAVA\Practice\35> javac –d . .\TestExtInterface.java

PS D:\MCA\MCA SEM 3\JAVA\Practice\35> java extInterface.TestExtInterface

Implementation of Method1

Enter number to find the Square root in java:

9

Square root of number: 9.000000 is -> 3.000000

Q. Package BasicOperation

➔ Math.java

```
package basicoperation;
public interface Math {
    public void add();
    public void sub();
    public void mul();
    public void div();
}
```

➔ Student.java

```
package basicoperation;
import java.util.Scanner;
public class Student implements math{
    @Override
    public void add() {
        Scanner kb = new Scanner(System.in);
        System.out.println("Enter any two integer values to perform addition : ")
;
        int a = kb.nextInt();
        int b = kb.nextInt();
        int s = a + b;
        System.out.println("Diffrence of "+a+" and "+b+" is "+s);
    }

    @Override
    public void sub() {
        Scanner kb = new Scanner(System.in);
        System.out.println("Enter any two integer values to perform subtraction :
 ");
        int a = kb.nextInt();
        int b = kb.nextInt();
        int s = a - b;
        System.out.println("Diffrence of "+a+" and "+b+" is "+s);
    }
```

```java
    @Override
    public void mul() {
        Scanner kb = new Scanner(System.in);
        System.out.println("Enter any two integer values to perform multiplicatio
n : ");
        int a = kb.nextInt();
        int b = kb.nextInt();
        int s = a * b;
        System.out.println("Diffrence of "+a+" and "+b+" is "+s);
    }

    @Override
    public void div() {
        Scanner kb = new Scanner(System.in);
        System.out.println("Enter any two integer values to perform divition : ")
;
        int a = kb.nextInt();
        int b = kb.nextInt();
        int s = a / b;
        System.out.println("Diffrence of "+a+" and "+b+" is "+s);
    }
    public static void main(String[] args){
        Student stu = new Student();
        stu.add();
        stu.sub();
        stu.mul();
        stu.div();
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java basicoperation.Student

Enter any two integer values to perform addition :

10 20

Diffrence of 10 and 20 is 30

Enter any two integer values to perform subtraction :

30 20

Diffrence of 30 and 20 is 10

Enter any two integer values to perform multiplication :

2

30

Diffrence of 2 and 30 is 60

Enter any two integer values to perform divition :

40 2

Diffrence of 40 and 2 is 20

Q. Package mca

➔ **Solution.java**

```java
package mca;
public interface Solution{
    public void Hello();
    public void Welcome();
}
```

➔ **classA.java**

```java
package mca;
public class classA implements Solution {
    public void Hello(){
        System.out.println("Hello World");
    }
    public void Welcome(){
        System.out.println("Welcome  MCAIII");
    }

    public static void main(String[] args) {
        classA test = new classA();
        test.Hello();
        test.Welcome();
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java mca.classA

Hello World

Welcome MCAIII

Q. Package book

➔ **Books.java**

```java
package bookpack;

public class Books{
    //these are protected members
    protected String title;
    protected String author;
    protected int pubDate;

    public Books(String t, String a, int d)
    {
        title = t;
        author = a;
        pubDate = d;
    }

    public void show()
    {
        System.out.println(title);
        System.out.println(author);
        System.out.println(pubDate);
        System.out.println();
    }
}
```

➔ **UseBook.java**

```java
package bookpackext;
import bookpack.*;

class UseBook {
    public static void main(String args[])
    {
        Books book[] = new Books[5];
        book[0] = new Books("Java: A Beginner's Guide","Schildt",2014);

        book[1] = new Books("Java: The Complete Reference","Schildt",2014);

        book[2] = new Books("The Art of Java","Schildt and Holmes",2003);
```

```
        book[3] = new Books("Red Storm Raising","Clancy",1986);

        book[4] = new Books("On The Road","Kerouac",1955);

        for(int index = 0; index < book.length;index++) book[index].show();
    }
}
```

→ **ProtectDemo.java**

```
package bookpackext;

class ExtBook extends bookpack.Books{
    private String publisher;
    public ExtBook(String t,String a, int d,String p)
    {
        super(t,a,d);
        publisher = p;
    }

    public void show()
    {
        super.show();
        System.out.println(publisher);
        System.out.println();
    }

    public String getPublisher() { return publisher; }
    public void setPublisher(String p) { publisher = p; }

    public String getTitle() { return title; }
    public void setTitle(String t) { title = t; }

    public String getAuthor() { return author; }
    public void setAuthor(String a) { author = a; }

    public int getPubDate() { return pubDate; }
    public void setTitle(int d) { pubDate = d; }

}
```

```java
class ProtectDemo
{
    public static void main(String[] args)
    {
        ExtBook books[] = new ExtBook[5];
        books[0] = new ExtBook("Java: A Beginner's Guide","Schildt",2014,"McGrew-
Hill Professional");

        books[1] = new ExtBook("Java: The Complete Reference","Schildt",2014,"McG
rew-Hill Professional");

        books[2] = new ExtBook("The Art of Java","Schildt and Holmes",2003,"McGre
w-Hill Professional");

        books[3] = new ExtBook("Red Storm Raising","Clancy",1986,"Putham");

        books[4] = new ExtBook("On The Road","Kerouac",1955,"Viking");

        for(int index = 0; index < books.length; index++)
        {
            books[index].show();
        }

        System.out.println("Showing all books by schildt.");
        for(int index=0;index<books.length;index++)
        {
            if(books[index].getAuthor() == "Schildt")
            {
                System.out.println(books[index].getTitle());
            }
            //books[index].title = "test title"; error by accessing protected
        }
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> javac -d . .\Books.java

PS D:\MCA\MCA SEM 3\JAVA\Practice> javac -d . .\UseBook.java

PS D:\MCA\MCA SEM 3\JAVA\Practice> javac -d . .\ProtectDemo.java

PS D:\MCA\MCA SEM 3\JAVA\Practice> java bookpackext.ProtectDemo

Java: A Beginner's Guide

Schildt

2014

McGrew-Hill Professional

Java: The Complete Reference

Schildt

2014

McGrew-Hill Professional

The Art of Java

Schildt and Holmes

2003

McGrew-Hill Professional

Red Storm Raising

Clancy

1986

Putham

On The Road

Kerouac

1955

Viking

Showing all books by schildt.

Java: A Beginner's Guide

Java: The Complete Reference

Q. Series Program

➔ Series.java

```java
public interface Series
{
    int getNext();
    void reset();
    void setStart(int x);
}
```

➔ SeriesDemo.java

```java
class ByTwos implements Series
{
    int start;
    int val;

    ByTwos() {
        start = 0;
        val = 0;
    }

    public int getNext() {
        val += 2;
        return val;
    }

    public void reset()
    {
        val = start;
    }

    public void setStart(int x)
    {
        start = x;
        val = x;
    }
}

class ByThrees implements Series
{
    int start;
```

```java
    int val;

    ByThrees() {
        start = 0;
        val = 0;
    }

    public int getNext(){
        val += 3;
        return val;
    }

    public void reset()
    {
        val = start;
    }

    public void setStart(int x)
    {
        start = x;
        val = x;
    }
}

class SeriesDemo{
    public static void main(String args[])
    {
        ByTwos twoOb = new ByTwos();
        ByThrees threeOb = new ByThrees();
        Series ob;

        for(int i=0; i < 5; i++)
        {
            ob = twoOb;
            System.out.println("Next ByTwos value is " +ob.getNext());

            ob = threeOb;
            System.out.println("Next ByThrees value is " + ob.getNext());
        }
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> javac -d . .\Series.java

PS D:\MCA\MCA SEM 3\JAVA\Practice> javac -d . .\SeriesDemo.java

PS D:\MCA\MCA SEM 3\JAVA\Practice> java SeriesDemo

Next ByTwos value is 2

Next ByThrees value is 3

Next ByTwos value is 4

Next ByThrees value is 6

Next ByTwos value is 6

Next ByThrees value is 9

Next ByTwos value is 8

Next ByThrees value is 12

Next ByTwos value is 10

Next ByThrees value is 15

Q. Lambda Displayable

```java
public class LambdaDisplayable implements displayble {
    public void display(String msg){
        System.out.println(msg);
    }
    public static void main(String[] args) {
        LambdaDisplayable dis = new LambdaDisplayable();
        dis.display("Welcome to the World.");

    }

}
interface displayble {
    void display(String  msg);

}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\LambdaDisplayable.java

Welcome to the World.

Q. Lambda with functional Interface

```java
public class LambdaFuntionalInterface {
    public static void main(String[] args)
    {
        FuncInterface fobj  = (int x) -> System.out.println(2*x);
        fobj.abstractFun(12);
    }
}

interface FuncInterface{
    void abstractFun(int x);

    default void normalFun()
    {
        System.out.println("Hello");
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\LambdaFuntionalInterface.java

24

Q. Calculator Using Lambda

```java
public class FunctionalLambda {
    interface IntegerMath {
        int operation(int a, int b);
    }

    public int operateBinary(int a, int b, IntegerMath op) {
        return op.operation(a, b);
    }

    public static void main(String[] args) {
        FunctionalLambda myApp = new FunctionalLambda();

        IntegerMath addition = (a, b) -> a + b;
        IntegerMath subtraction = (a, b) -> a - b;
        IntegerMath multiplication = (a, b) -> a * b;
        IntegerMath division = (a, b) -> a / b;

        int a = 10, b = 5;

        System.out.println(a + " + " + b + " = " + myApp.operateBinary(a, b, addition));
        System.out.println(a + " -
 " + b + " = " +myApp.operateBinary(a, b, subtraction));
        System.out.println(a + " * " + b + " = " + myApp.operateBinary(a, b, multiplication));
        System.out.println(a + " / " + b + " = " + myApp.operateBinary(a, b, division));
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\FunctionalLambda.java

10 + 5 = 15

10 - 5 = 5

10 * 5 = 50

10 / 5 = 2

Q. Implement Uber using interface, Anonymous class and Lambda Expression

➔ **Uber1.java**

```java
public class Uber1 {
    public static void main(String[] args) {
        Cab cab = new UberX1();
        cab.bookCab();
    }
}

@FunctionalInterface
interface Cab {
    void bookCab();
}

class UberX1 implements Cab {
    public void bookCab() {
        System.out.println("UberX Booked!! Arriving Soon !!");
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\Uber1.java

UberX Booked!! Arriving Soon !!

➔ **Uber2.java**

```java
public class Uber2 {
    public static void main(String[] args) {
        Cab cab = new Cab() {

            @Override
            public void bookCab() {
                System.out.println("UberX Booked!! Arriving Soon !!");
            }
        };

        cab.bookCab();
```

```
        }
}

@FunctionalInterface
interface Cab {
    void bookCab();
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\Uber2.java

UberX Booked!! Arriving Soon !!

➔ **Uber3.java**

```
public class Uber3 {
    public static void main(String[] args) {
        Cab cab = () -> System.out.println("UberX Booked!! Arriving Soon !!");
        cab.bookCab();
    }
}

@FunctionalInterface
interface Cab {
    void bookCab();
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\Uber3.java

UberX Booked!! Arriving Soon !!

Practical Assignments - 1

Q. First 50 Prime Numbers

```java
public class For50Prime {
    public static void main(String[] args) {
        System.out.println("50 Prime Numbers using For Loop");
        ForPrime();
        System.out.println();
        System.out.println("50 Prime Numbers using While Loop");
        WhilePrime();
        System.out.println();
        System.out.println("50 Prime Numbers using Do-While Loop");
        DoWhilePrime();
    }


    public static void ForPrime() {
        boolean checkPrime = false;
        int count = 1;

        for(int i = 2; count < 51; i++){
            int sqroot = (int)Math.sqrt(i);
            for(int j = 2; j <= sqroot; j++)
            {
                if(i % j == 0)
                {
                    checkPrime = true;
                    break;
                }
            }
            if(!checkPrime){
                System.out.println(count + " = " + i);
                count++;
            }
            else
                checkPrime = false;
        }
    }
    public static void WhilePrime() {
        boolean checkPrime = false;
        int count = 1;
        int i = 2;
```

```java
        while(count < 51) {
            int sqroot = (int)Math.sqrt(i);
            int j = 2;
            while(j <= sqroot)
            {
                if(i % j == 0)
                {
                    checkPrime = true;
                    break;
                }
                j++;
            }
            if(!checkPrime){
                System.out.println(count + " = " + i);
                count++;
            }
            else checkPrime = false;

            i++;
        }
    }

    public static void DoWhilePrime() {
        boolean checkPrime = false;
        int count = 1;
        int i = 2;
        do{
            int sqroot = (int)Math.sqrt(i);
            int j = 2;
            while(j <= sqroot) {
                if(i % j == 0)
                {
                    checkPrime = true;
                    break;
                }
                j++;
            }
            if(!checkPrime){
                System.out.println(count + " = " + i);
                count++;
            }
            else checkPrime = false;

            i++;
        }while(count < 51);
    }

}
```

Output :

50 Prime Numbers using For Loop

1 = 2

2 = 3

3 = 5

4 = 7

5 = 11

6 = 13

7 = 17

8 = 19

9 = 23

10 = 29

11 = 31

12 = 37

13 = 41

14 = 43

15 = 47

16 = 53

17 = 59

18 = 61

19 = 67

20 = 71

21 = 73

22 = 79

23 = 83

24 = 89

25 = 97

26 = 101

27 = 103

28 = 107

29 = 109

30 = 113

31 = 127

32 = 131

33 = 137

34 = 139

35 = 149

36 = 151

37 = 157

38 = 163

39 = 167

40 = 173

41 = 179

42 = 181

43 = 191

44 = 193

45 = 197

46 = 199

47 = 211

48 = 223

49 = 227

50 = 229

50 Prime Numbers using While Loop

1 = 2

2 = 3

3 = 5

4 = 7

5 = 11

6 = 13

7 = 17

8 = 19

9 = 23

10 = 29

11 = 31

12 = 37

13 = 41

14 = 43

15 = 47

16 = 53

17 = 59

18 = 61

19 = 67

20 = 71

21 = 73

22 = 79

23 = 83

24 = 89

25 = 97

26 = 101

27 = 103

28 = 107

29 = 109

30 = 113

31 = 127

32 = 131

33 = 137

34 = 139

35 = 149

36 = 151

37 = 157

38 = 163

39 = 167

40 = 173

41 = 179

42 = 181

43 = 191

44 = 193

45 = 197

46 = 199

47 = 211

48 = 223

49 = 227

50 = 229

50 Prime Numbers using Do-While Loop

1 = 2

2 = 3

3 = 5

4 = 7

5 = 11

6 = 13

7 = 17

8 = 19

9 = 23

10 = 29

11 = 31

12 = 37

13 = 41

14 = 43

15 = 47

16 = 53

17 = 59

18 = 61

19 = 67

20 = 71

21 = 73

22 = 79

23 = 83

24 = 89

25 = 97

26 = 101

27 = 103

28 = 107

29 = 109

30 = 113

31 = 127

32 = 131

33 = 137

34 = 139

35 = 149

36 = 151

37 = 157

38 = 163

39 = 167

40 = 173

41 = 179

42 = 181

43 = 191

44 = 193

45 = 197

46 = 199

47 = 211

48 = 223

49 = 227

50 = 229

Q. Multiply and Display Product

```java
public class Product {
    public static void main(String args[]){
        int i = 0;
        double total = 1;
        while(i < args.length){
            total *= Double.parseDouble(args[i]);
            i++;
        }
        System.out.println("Product : " + total);
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Assignment 1> java .\Product.java 4.5 3.2

Product : 14.4

Q. Area of circle

```java
public class AreaOfCircle {
    public static void main(String[] args)
    throws java.io.IOException{
        double pi = 3.14159;
        System.out.print("Enter The Radius : ");
        double r = (double) System.in.read() - 48;
        System.out.println("Area Of Circle : " + pi*Math.pow(r,2));
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Assignment 1> java .\AreaOfCircle.java

Enter The Radius : 5

Area Of Circle : 78.53975

Q. Display Milliseconds between current date and midnight.

```java
import java.util.Calendar;
public class Milliseconds {
    final static byte HOURS = 24;
    final static byte MINUTE = 60;
    final static byte SECONDS = 60;
    final static short MILLISECONDS = 1000;
    public static void main(String[] args) {

        Calendar date = Calendar.getInstance();
        int currentHour = date.get(Calendar.HOUR_OF_DAY);
        int currentMinute = date.get(Calendar.MINUTE);
        int currentSecond = date.get(Calendar.SECOND);
        int currentMillisecond = date.get(Calendar.MILLISECOND);
        System.out.println("Current Time : " + currentHour + ":" + currentMinu
te + ":" + currentSecond + ":" + currentMillisecond);

        long tillMidnightMilliseconds = ( (HOURS - currentHour - 1) * MINUTE *
 SECONDS * MILLISECONDS ) + ((MINUTE - currentMinute ) * SECONDS * MILLISECOND
S) + ((SECONDS - currentSecond ) * MILLISECONDS) + (MILLISECONDS - currentMill
isecond);
        System.out.println("Milliseconds Remain till Midnight: " + tillMidnigh
tMilliseconds);
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Assignment 1> java .\Milliseconds.java

Current Time : 18:20:27:590

Milliseconds Remain till Midnight: 20433410

Q. Square Root

```java
public class SquareRoot {
    public static void main(String args[]){
        System.out.println(Math.sqrt(Double.parseDouble(args[0])));
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Assignment 1> java .\SquareRoot.java 5

2.23606797749979

Q. Display Numbers between 17 to 100 which evenly divisible by 17

```java
public class Print17 {
    public static void main(String[] args){
        int i = 1;
        int magicNumber = 17;

        while((magicNumber*i) <= 100){
            System.out.print((magicNumber*i) + " ");
            i++;
        }
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Assignment 1> java .\Print17.java

17 34 51 68 85

Q. Possible Factors

```java
public class PosibleFactors {
    public static void main(String args[]){
        int number = Integer.parseInt(args[0]);
        int i = 1;
        while(i <= (number/2)){
            if( number % i == 0 )
                System.out.print(i + " ");

            i++;
        }
        System.out.print(number);
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Assignment 1> java .\PosibleFactors.java 89

1 89

PS D:\MCA\MCA SEM 3\JAVA\Assignment 1> java .\PosibleFactors.java 98

1 2 7 14 49 98

Q. Five Columns

```java
public class PosibleFactors {
    public static void main(String args[]){
        int number = Integer.parseInt(args[0]);
        int i = 1;
        while(i <= (number/2)){
            if( number % i == 0 )
                System.out.print(i + " ");

            i++;
        }
        System.out.print(number);
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Assignment 1> java .\FiveColumn.java

1 2 3 4 5

6 7 8 9 10

11 12 13 14 15

16 17 18 19 20

21 22 23 24 25

26 27 28 29 30

31 32 33 34 35

36 37 38 39 40

41 42 43 44 45

46 47 48 49 50

51 52 53 54 55

56 57 58 59 60

61 62 63 64 65

66 67 68 69 70

71 72 73 74 75

76 77 78 79 80

81 82 83 84 85

86 87 88 89 90

91 92 93 94 95

96 97 98 99 100

Q. Pass Number as argument and decrement it & when it reaches 0 then sound bell.

```java
public class Bell {
    public static void main(String args[]) {
        int bellCount = Integer.parseInt(args[0]);

        while(bellCount != 0){
            System.out.println(bellCount);
            bellCount--;
        }
        System.out.print("\u0007");
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Assignment 1> java .\Bell.java 7

7

6

5

4

3

2

1

Q. Class Person

```java
import java.util.Scanner;
public class Person {
    private String name;
    private int age;
    private float salary;

    public void set(){
        Scanner get = new Scanner(System.in);
        System.out.print("Enter Name : ");
        name = get.nextLine();
        System.out.print("Enter Age : ");
        age = get.nextInt();
        System.out.print("Enter Salary : ");
        salary = get.nextFloat();
    }

    public void display(){
        System.out.println("Name : " + name);
        System.out.println("Age : " + age);
        System.out.println("Salary : " + salary);
    }

    public static void main(String[] args){
        Person person = new Person();
        person.set();
        person.display();
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Assignment 1> java .\Person.java

Enter Name : Pradip

Enter Age : 22

Enter Salary : 30000

Name : Pradip

Age : 22

Salary : 30000.0

Practical Assignments - 2

Q. Add Matrices

```java
import java.util.Scanner;
public class MatrixAddition {
    public static void main(String[] args){
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter Number Of Rows : ");
        byte row = scan.nextByte();
        System.out.print("Enter Number Of Columns : ");
        byte col = scan.nextByte();
        int[][] matrixOne = new int[row][col];
        int[][] matrixTwo = new int[row][col];

        System.out.println("Matrix 1");
        getMatrix(matrixOne,row,col);
        System.out.println("Matrix 2");
        getMatrix(matrixTwo,row,col);
        addAndDisplayMatrix(matrixOne, matrixTwo, row, col);
    }

    private static void getMatrix(int[][] matrix,byte row,byte col) {
        Scanner scan = new Scanner(System.in);
        for (int i = 0; i < row; i++) {
            for (int j = 0; j < col; j++) {
                System.out.print("Enter Number for Matrix [" + i + "][" + j +
"] : ");
                matrix[i][j] = scan.nextInt();
            }
        }
    }

    private static void addAndDisplayMatrix(int[][] matrix1,int[][] matrix2,by
te row,byte col) {
        System.out.println("Final Matrix");
        for (int i = 0; i < row; i++) {
            for (int j = 0; j < col; j++) {
                System.out.print(matrix1[i][j]+matrix2[i][j] + " ");
            }
            System.out.println();
        }
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Assignment 2> java .\MatrixAddition.java

Enter Number Of Rows : 3

Enter Number Of Columns : 3

Matrix 1

Enter Number for Matrix [0][0] : 5

Enter Number for Matrix [0][1] : 4

Enter Number for Matrix [0][2] : 3

Enter Number for Matrix [1][0] : 6

Enter Number for Matrix [1][1] : 3

Enter Number for Matrix [1][2] : 8

Enter Number for Matrix [2][0] : 1

Enter Number for Matrix [2][1] : 3

Enter Number for Matrix [2][2] : 5

Matrix 2

Enter Number for Matrix [0][0] : 3

Enter Number for Matrix [0][1] : 2

Enter Number for Matrix [0][2] : 5

Enter Number for Matrix [1][0] : 6

Enter Number for Matrix [1][1] : 4

Enter Number for Matrix [1][2] : 7

Enter Number for Matrix [2][0] : 1

Enter Number for Matrix [2][1] : 0

Enter Number for Matrix [2][2] : 2

Final Matrix

8 6 8

12 7 15

2 3 7

Q. Get Environment Variables

```java
import java.lang.*;
public class GetEnvironmentVariable {
    public static void main(String[] args)
    {
        System.out.println("\nEnvironment variable USERNAME: ");
        System.out.println(System.getenv("USERNAME"));

        System.out.println("\nEnvironment variable TEMP: ");
        System.out.println(System.getenv("TEMP"));

        System.out.println("\nEnvironment variable PATH: ");
        System.out.println(System.getenv("PATH"));

    }
}
```

Output:

PS D:\MCA\MCA SEM 3\JAVA\Assignment 2> java .\GetEnvironmentVariable.java

Environment variable USERNAME:

Xtrem

Environment variable TEMP:

C:\Users\Xtrem\AppData\Local\Temp

Environment variable PATH:

C:\Windows\system32;C:\Windows;C:\Windows\System32\Wbem;C:\Windows\System32\WindowsPowerShell\v1.0\;C:\Windows\System32\OpenSSH\;C:\Program Files (x86)\NVIDIA Corporation\PhysX\Common;C:\Program Files\NVIDIA Corporation\NVIDIA NvDLISR;C:\Program Files\Java\jdk-13.0.1\bin;C:\Program Files\Git\cmd;C:\Users\Xtrem\AppData\Local\Microsoft\WindowsApps;C:\Users\Xtrem\AppData\Local\Programs\Microsoft VS Code\bin;C:\Program Files\Java\jdk-13.0.1\bin;

Q. Circle

```java
public class Circle{
    public static void main(String[] args)
    {
        double x,y,r;
        int origin;
        CircleObj[] c= new CircleObj[10];

        for(int i=0;i<10;i++)
        {
            origin=(int) (2*  Math.random());
            r= (double) (10* Math.random())+1;

            if(origin==1)
                c[i]=new CircleObj(r);
            else
            {
                x= (double) (100* Math.random())+1;
                y= (double) (100* Math.random())+1;
                c[i] = new CircleObj(x,y,r);
            }

        }
        for(int i=0;i<10;i++)
        {
            c[i].display();
        }
    }
}
public class CircleObj
{
    double radius,cntx,cnty;

    CircleObj(double radius)
    {
        this.cntx=0;
        this.cnty=0;
        this.radius=radius;
    }

    CircleObj(double cntx,double cnty,double radius)
    {
        this.cntx=cntx;
        this.cnty=cnty;
        this.radius=radius;
    }
```

```java
    private double area()
    {
        return (3.14*this.radius*this.radius);
    }

    public void display()
    {
         System.out.println("-------");
        System.out.println("Center: ("+this.cntx+", "+this.cnty+")");
        System.out.println("Radius: "+this.radius);
        System.out.println("Area: "+ this.area());
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Assignment 2> java .\Circle.java

Center: (71.78505496544743, 17.213464081660057)

Radius: 10.101627819491677

Area: 320.4146576550785

Center: (0.0, 0.0)

Radius: 5.0617069059482045

Area: 80.44955315741257

Center: (0.0, 0.0)

Radius: 10.362498572521107

Area: 337.1775227296762

Center: (89.67442172582656, 67.80023035725446)

Radius: 4.762193006142564

Area: 71.21043419514488

Center: (47.914239717727405, 25.360860928995752)

Radius: 5.406426040616273

Area: 91.78044955253277

Center: (0.0, 0.0)

Radius: 10.088821343762639

Area: 319.6027525739722

Center: (0.0, 0.0)

Radius: 10.6691310947075

Area: 357.4273251124111

Center: (0.0, 0.0)

Radius: 1.1219839322884109

Area: 3.9527825451439673

Center: (0.0, 0.0)

Radius: 3.6632745450908812

Area: 42.13748243311192

Center: (0.0, 0.0)

Radius: 3.817459601092299

Area: 45.759213110751375

Q. Remove duplicate characters from the string.

```java
import java.util.Arrays;
public class RemoveDuplicate {
    public static void main(String[] args) {
        char[] myString = "Pradip Karmakar".toCharArray();
        System.out.println(StringSlash(myString));
    }

    private static String StringSlash(char[] myString) {
        int index = 0;
        int j;
        for (int i = 0; i < myString.length; i++) {

            for (j = 0; j < i; j++) {
                if (myString[i] == myString[j])
                {
                    break;
                }
            }

            if(j == i){
                myString[index++] = myString[i];
            }
        }
        return String.valueOf(Arrays.copyOf(myString, index));
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Assignment 2> java .\RemoveDuplicate.java

Pradip Kmk

Q. Write a program to print the pattern using for loop

1 1 1

2 4 8

....

10 100 1000

```java
public class SquareCube {
    public static void main(String[] args){
        for (int i = 1; i < 11; i++) {
            for (int j = 1; j < 4; j++) {
                System.out.print((int)Math.pow(i, j) + "  ");
            }
            System.out.println();
        }
    }
}
```

Output:

PS D:\MCA\MCA SEM 3\JAVA\Assignment 2> java .\SquareCube.java

1 1 1

2 4 8

3 9 27

4 16 64

5 25 125

6 36 216

7 49 343

8 64 512

9 81 729

10 100 1000

Q. Write a java program to accept distance in meters, time in hours ,minutes and seconds .Calculate the speed in m/sec ,km/hr and miles/hr .(1 mile=1609 m).

```java
import java.util.Scanner;
public class Speed {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);

        float seconds;
        float minutePerSecond;
        float kiloMeterPerHour;
        float milePerHour;

        System.out.print("Enter distance in meters : ");
        float distance = scan.nextFloat();

        System.out.print("Enter hour: ");
        float hr = scan.nextFloat();

        System.out.print("Enter minutes: ");
        float min = scan.nextFloat();

        System.out.print("Enter seconds: ");
        float sec = scan.nextFloat();

        seconds = (hr*3600) + (min*60) + sec;
        minutePerSecond = distance / seconds;
        kiloMeterPerHour = ( distance/1000.0f ) / ( seconds/3600.0f );
        milePerHour = kiloMeterPerHour / 1.609f;

        System.out.println("Meter/Seconds : " + minutePerSecond);
        System.out.println("KM/H : " + kiloMeterPerHour);
        System.out.println("Miles / Hour : " + milePerHour);

    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Assignment 2> java .\Speed.java

Enter distance in meters : 80000

Enter hour: 1

Enter minutes: 30

Enter seconds: 45

Meter/Seconds : 14.692378

KM/H : 52.89256

Miles / Hour : 32.87294

Q. Display Java detail

```
public class JavaInfo {
    public static void main(String[] args){
        System.out.println("Java Version : " + System.getProperty("java.versio
n"));
        System.out.println("Runtime Version : " + System.getProperty("java.run
time.version"));
        System.out.println("Java Vendor Name : " + System.getProperty("java.ve
ndor"));
        System.out.println("Java Vendor URL : " + System.getProperty("java.ven
dor.url"));
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Assignment 2> java .\JavaInfo.java

Java Version : 13.0.1

Runtime Version : 13.0.1+9

Java Vendor Name : Oracle Corporation

Java Vendor URL : https://java.oracle.com/

Q. Create an abstract class "Monster" which is extended by three classes "Ware wolf ","Zombie" and "Vampire" .Create 6 types of Monsters in single dimension array and print them.

```java
public class MonstersObject
{
    public static void main(String[] args)
    {
        int key;
        Monster[] m= new Monster[6];

        for(int i=0;i<6;i++)     //to create
        {
            key=(int) (3 * Math.random()) +1;    //to create random Monsters

            switch(key)
            {
                case 1: //For Wolf
                    m[i]= new WareWolf(i);
                    break;
                case 2: //for Zombie
                    m[i]= new Zombie(i);
                    break;
                case 3: //for Vampire
                    m[i]= new Vampire(i);
            }
        }

        for(int i=0;i<6;i++)
        {
            m[i].display();
        }
    }
}
public class MonsterObject {

}
abstract class Monster
{
    String type;
    int id;

    Monster(String type)
    {
        this.type=type;
    }
    public void display()
    {
```

```java
            System.out.println("-----");
            System.out.println("Monster id: "+ this.id);
            System.out.println("Monster type: "+ this.type);
    }
}

class WareWolf extends Monster
{
    WareWolf(int id)
    {
        super("WareWolf");
        this.id=id;
    }
}

class Zombie extends Monster
{
    Zombie(int id)
    {
        super("Zombie");
        this.id=id;
    }
}
class Vampire extends Monster
{
    Vampire(int id)
    {
        super("Vampire");
        this.id=id;
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Assignment 2> java .\MonsterObject.java

-----

Monster id: 0

Monster type: Vampire

Monster id: 1

Monster type: WareWolf

Monster id: 2

Monster type: Vampire

Monster id: 3

Monster type: Zombie

Monster id: 4

Monster type: Vampire

Monster id: 5

Monster type: Vampire

Q. Write a program to accept a string from user and a point from where you want to print next three

words. Print the old and the new string.

Old String = "The quick brown fox jumps over the lazy dog"

New String = "brown fox jumps"

```java
import java.util.Scanner;

public class StringSplit {
    public static void main(String[] args)
    {
    int index;
    int tmp;
    int count=0;
    String old;
    String New;
    String token;
    Scanner s = new Scanner(System.in);

    System.out.print("Enter String: ");
    old=s.nextLine();

    System.out.print("Enter from which word you want to split: ");
    token=s.nextLine();

    index= old.indexOf(token);
    tmp=index;

    while(count<3 && tmp != -1)
    {
        tmp=old.indexOf(" ",tmp+1);
        count++;
    }

    New=old.substring(index,tmp);

    System.out.println("Old string= " + old);
    System.out.println("New string= " + New);


    }
}
```

Output:

PS D:\MCA\MCA SEM 3\JAVA\Assignment 2> java .\StringSplit.java

Enter String: Pradip Karmakar Study in Rollwala Computer Center

Enter from which word you want to split: Karmakar

Old string= Pradip Karmakar Study in Rollwala Computer Center

New string= Karmakar Study in

Q. Convert Minutes into years and days

For Eg. Input => 3456789

Output => 6 Years and 210 days

```java
import java.util.Scanner;

public class MinuteToYearDay {
    final static int HOUR = 24;
    final static int DAYSINYEAR = 365;
    final static int MINUTES = 60;
    public static void main(String[] args){
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter Values as Minutes : ");
        long minute = scan.nextLong();
        int year = (int) (minute / ((MINUTES*HOUR)*DAYSINYEAR));
        int remainingminute = (int) (minute % ((MINUTES*HOUR)*DAYSINYEAR));
        int days = remainingminute / (MINUTES * HOUR);

        System.out.println(year + " Years and " + days + " Days.");
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Assignment 2> java .\MinuteToYearDay.java

Enter Values as Minutes : 12000000

22 Years and 303 Days.

Q. Generate Invoice

```java
public class Invoice {
    public static void main(String[] args) {
        Inventory items[] = { new Inventory(101, "Part 1", 2000), new Inventory(102, "Part 2", 200),
                new Inventory(103, "Part 3", 1300), new Inventory(104, "Part 4", 3200), new Inventory(105, "Part 5", 5000) };

        Bill_Item bitem = new Bill_Item(items.length);
        for (int i = 0; i < items.length; i++) {
            bitem.add(items[i], 1);
        }
        bitem.display();
    }
}

class Inventory {
    long itemId;
    String description;
    double price;
    int qty;

    Inventory() {
        itemId = 0;
        description = "";
        price = 0;
    }

    Inventory(long itemId, String description, double price) {
        this.itemId = itemId;
        this. description = description;
        this.price = price;
    }

    public void display() {
        System.out.println("Item id : " + itemId);
        System.out.println("\t Description : " + description);
        System.out.println("\t Price " + price);
        System.out.println("\t qty " + qty);
        System.out.println("_____");
    }
}

class Bill_Item {

    int qty;
    double amount;
```

```java
    Inventory items[];

    int index = 0;

    Bill_Item(int length) {
        items = new Inventory[length];
        index = 0;
    }

    public void add(Inventory it, int qty) {
        this.amount = it.price * qty;
        it.qty += qty;
        items[index] = it;
        index++;
    }

    public void display() {
        int total = 0;
        for (int i = 0; i < items.length; i++) {
            items[i].display();
            total += items[i].qty * items[i].price;
            System.out.println();
        }

        System.out.print("Amount : " + total);
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Assignment 2> java .\Invoice.java

Item id : 101

Description : Part 1

Price 2000.0

qty 1

Item id : 102

Description : Part 2

Price 200.0

qty 1

Item id : 103

Description : Part 3

Price 1300.0

qty 1

Item id : 104

Description : Part 4

Price 3200.0

qty 1

Item id : 105

Description : Part 5

Price 5000.0

qty 1

Amount : 11700

Q. Count Words and Print in Reverse

```java
import java.util.Scanner;

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

        System.out.println("Enter String : ");
        input = scan.nextLine();

        String token[] = input.split(" ");
        System.out.println("Count of words in String : " + token.length);

        for (int i = token.length - 1; i >= 0; i--) {
            System.out.print(token[i] + " ");
        }
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Assignment 2> java .\WordReverse.java

Enter String :

Pradip Karmakar Study at Rollwala

Count of words in String : 5

Rollwala at Study Karmakar Pradip

Q. Binary Addition

```java
import java.util.Scanner;
public class BinaryAdd {
    public static void main(String[] args) {
        long b1, b2;
        int i = 0, carry = 0;

        int[] sum = new int[10];

        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter first binary number: ");
        b1 = scanner.nextLong();
        System.out.print("Enter second binary number: ");
        b2 = scanner.nextLong();

        scanner.close();
        while (b1 != 0 || b2 != 0)
        {
            sum[i++] = (int)((b1 % 10 + b2 % 10 + carry) % 2);
            carry = (int)((b1 % 10 + b2 % 10 + carry) / 2);
            b1 = b1 / 10;
            b2 = b2 / 10;
        }
        if (carry != 0) {
            sum[i++] = carry;
        }
        --i;
        System.out.print("Output: ");
        while (i >= 0) {
            System.out.print(sum[i--]);
        }
        System.out.print("\n");
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Assignment 2> java .\BinaryAdd.java

Enter first binary number: 1010

Enter second binary number: 1100

Output: 10110

Q. Print Java

```java
public class PrintJava {
    public static void main(String[] args) {
        System.out.println("   JJ      A     V         V      A");
        System.out.println("   JJ     A A     V      V     A A");
        System.out.println("   JJ    A   A    V    V     A   A");
        System.out.println("   JJ    AAAAAAA    V V     AAAAAAA");
        System.out.println(" JJJJJ  A       A    v    A         A");
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Assignment 2> java .\PrintJava.java

```
   JJ      A    V         V     A

   JJ     A  A   V        V   A   A

   JJ    A   A   V      V   A      A

   JJ   AAAAAAA    V   V   AAAAAAA

 JJJJJ  A          A     v     A              A
```

Q. Print Face

```java
public class PrintFace {
    public static void main(String[] args) {
        System.out.println(" +\"\"\"\"\"\"\"+ ");
        System.out.println("[| 0   0 |]");
        System.out.println(" |   ^   | ");
        System.out.println(" |  '_'  | ");
        System.out.println(" +-------+ ");
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Assignment 2> java .\PrintFace.java

+"""""""+

[| 0 0 |]

| ^ |

| '_' |

+----------+

Q. Print Pattern

```java
public class OneZeroPattern {
    public static void main(String[] args) {
        int setter = 1;
        int numberOflines = 9;
        for (int i = 0; i < numberOflines; i++) {
            for (int j = 0; j <= i; j++) {
                System.out.print(setter);
                if(setter == 1) setter = 0;
                else setter = 1;
            }
            System.out.println("");
        }
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Assignment 2> java .\OneZeroPattern.java 5

1

01

010

1010

10101

010101

0101010

10101010

101010101

Q. 14)Create an employee class which has attributes : empid , fname , lname, salary, designation, bonus. It has all the necessary constructors (at least 2) and methods.It must have one abstract method calculate_bonus().

```java
public class Data {
    public static void main(String[] args) {
        Employee emp1 = new Manager(101, "Pradip", "Karmakar", 30_00_0F, "Prod
uct Manager", 6);
        Employee emp2 = new Clerk(201, "Sudip", "Karmakar", 10_00_0F, "Product
ion", 300);
        Employee emp3 = new Manager(102, "Sandip", "Karmakar", 23_00_0F, "Sale
s Manager", 2);
        Employee emp4 = new Clerk(202, "Jeet", "Karmakar", 30_00_0F, "Sales",
240);
        Employee emp5 = new Manager(103, "Samar", "Karmakar", 40_00_0F, "Human
 Resource Manager", 12);
        emp1.CalculateBonus();
        emp1.PrintDetail();
        emp2.CalculateBonus();
        emp2.PrintDetail();
        emp3.CalculateBonus();
        emp3.PrintDetail();
        emp4.CalculateBonus();
        emp4.PrintDetail();
        emp5.CalculateBonus();
        emp5.PrintDetail();
    }
}

abstract class Employee {
    int empid;
    String fname;
    String lname;
    float salary;
    String designation;
    float bonus;

    public Employee() {
        empid = 0;
        fname = "Not Specified";
        lname = "Not Specified";
        salary = 0.0F;
        designation = "Not Specified";
```

```java
        bonus = 0.0F;
    }

    public Employee(int empid, String fname, String lname, float salary, String designation) {
        this.empid = empid;
        this.fname = fname;
        this.lname = lname;
        this.salary = salary;
        this.designation = designation;
    }

    public abstract void CalculateBonus();
    public abstract void PrintDetail();
}

class Manager extends Employee {
    int noOfProjectsHandled;

    Manager() {
        super();
    }
    Manager(int empid, String fname, String lname, float salary, String designation, int noOfProjectsHandled) {
        super(empid, fname, lname, salary, designation);
        this.noOfProjectsHandled = noOfProjectsHandled;
    }

    public void CalculateBonus() {
            this.bonus = (noOfProjectsHandled * 1000);
    }

    public void PrintDetail() {
        System.out.println("Employee Id : " + this.empid);
        System.out.println("Name : " + this.fname + " " + this.lname);
        System.out.println("Salary : "+ this.salary);
        System.out.println("Designation : " + this.designation);
        System.out.println("Total Projects Handeled : " + this.noOfProjectsHandled);
        System.out.println("Bonus : " + this.bonus);
        System.out.println("++++++++++++++++++++++++");
    }
}

class Clerk extends Employee {
    int noOfHoursworked;

    Clerk() {
```

```java
        super();
    }
    Clerk(int empid, String fname, String lname, float salary, String designat
ion, int noOfHoursworked) {
        super(empid, fname, lname, salary, designation);
        this.noOfHoursworked = noOfHoursworked;
    }

    public void CalculateBonus() {
        if(this.noOfHoursworked > 250) {
            this.bonus = (noOfHoursworked - 250) * 200;
        }
    }

    public void PrintDetail() {
        System.out.println("Employee Id : " + this.empid);
        System.out.println("Name : " + this.fname + " " + this.lname);
        System.out.println("Salary : "+ this.salary);
        System.out.println("Designation : " + this.designation);
        System.out.println("Total Hours Worked : " + this.noOfHoursworked);
        System.out.println("Bonus : " + this.bonus);
        System.out.println("+++++++++++++++++++++++");
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Assignment 2> java .\Data.java

Employee Id : 101

Name : Pradip Karmakar

Salary : 30000.0

Designation : Product Manager

Total Projects Handeled : 6

Bonus : 6000.0

+++++++++++++++++++++++

Employee Id : 201

Name : Sudip Karmakar

Salary : 10000.0

Designation : Production

Total Hours Worked : 300

Bonus : 10000.0

+++++++++++++++++++++++

Employee Id : 102

Name : Sandip Karmakar

Salary : 23000.0

Designation : Sales Manager

Total Projects Handeled : 2

Bonus : 2000.0

+++++++++++++++++++++++

Employee Id : 202

Name : Jeet Karmakar

Salary : 30000.0

Designation : Sales

Total Hours Worked : 240

Bonus : 0.0

+++++++++++++++++++++++

Employee Id : 103

Name : Samar Karmakar

Salary : 40000.0

Designation : Human Resource Manager

Total Projects Handeled : 12

Bonus : 12000.0

+++++++++++++++++++++++

Class Work - 4

Q. ExcDemo1

```java
class ExcDemo1 {
    public static void main(String[] args) {
        int[] numbers = new int[9];
        try {
            System.out.println("Before Exception");
            numbers[10] = 200;
            System.out.println("this won't be displayed");
        }
        catch(ArrayIndexOutOfBoundsException exc) {
            System.out.println("index out-of bound !");
        }
        System.out.println("After catch statement");
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\ExcDemo1.java

Before Exception

index out-of bound !

After catch statement

Q. ExcDemo2

```java
public class ExcDemo2 {
    public static void main(String[] args) {
        int[] numberSet1 = { 12, 14, 12, 6, 18, 40 };
        int[] numberSet2 = { 2, 4, 0, 6, 0, 4 };
        try {
            System.out.println("Before Exception");
            for (int i = 0; i < numberSet1.length; i++) {
                System.out.println(numberSet1[i] / numberSet2[i]);
            }
            System.out.println("this won't be displayed");
        }
        catch(ArithmeticException exc) {
            System.out.println("Divide by Zero !");
        }
        System.out.println("After catch statement");
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\ExcDemo2.java

Before Exception

6

3

Divide by Zero !

After catch statement

Q. Finally Demo

```java
public class FinallyDemo {
    public static void main(String[] args) {
        for (int i = 0; i < 3; i++) {
            UseFinally.genException(i);
            System.out.println();
        }
    }
}

class UseFinally {
    public static void genException(int what) {
        int t;
        int nums[] = new int[2];
        System.out.println("Receiveing " + what);
        try {
            switch (what) {
                case 0:
                    t = 10 / what;
                    break;
                case 1:
                    nums[4] = 4;
                    break;
                case 2:
                    return;
            }
        } catch (ArithmeticException e) {
            System.out.println(e.getMessage());
        } catch (ArrayIndexOutOfBoundsException e) {
            System.out.println(e.getMessage());
        } finally {
            System.out.println("Leaving Try");
        }
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\FinallyDemo.java

Receiveing 0

/ by zero

Leaving Try

Receiveing 1

Index 4 out of bounds for length 2

Leaving Try

Receiveing 2

Leaving Try

Q. ThrowsDemo

```java
import java.io.IOException;

public class ThrowsDemo {
    public static char prompt(String str) throws IOException {
        System.out.print(str + " : ");
        return (char) System.in.read();
    }

    public static void main(String[] args) {
        char ch;

        try {
            ch = prompt("Enter a letter");
        } catch (IOException e) {
            System.out.println("I/O Exception occoured");
            ch = 'X';
        }
        System.out.println("You pressed " + ch);
    }
}
```

Output:

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\ThrowsDemo.java

Enter a letter : R

You pressed R

Q. MultiCatch

```java
public class MultiCatch {
    public static void main(String[] args) {
        int a = 88, b = 0;
        int result;
        char ch[] = { 'A', 'B', 'C' };

        for (int i = 0; i < 2; i++) {
            try {
                if (i == 0)
                    result = a / b;
                else
                    ch[5] = 'X';
            } catch (ArithmeticException | ArrayIndexOutOfBoundsException e) {
                System.out.println(e.getMessage());
            }
        }
        System.out.println("After multi Catch");
    }
}
```

Output:

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\MultiCatch.java

/ by zero

Index 5 out of bounds for length 3

After multi Catch

Q. Custom ExceptionsDemo

```java
public class CustomExceptionDemo {
    public static void main(String[] args) {
        int numer[] = { 4, 8, 15, 32, 64, 128, 256, 512 };
        int denom[] = { 2, 0, 4, 4, 0, 8 };

        for (int i = 0; i < numer.length; i++) {
            try {
                if (numer[i] % 2 != 0)
                    throw new NonIntResultException(numer[i], denom[i]);

                System.out.println(numer[i] + " / " + denom[i] + " is " + nume
r[i] / denom[i]);
            } catch (ArithmeticException e) {
                System.out.println("Cannot divide by Zero");
            } catch (ArrayIndexOutOfBoundsException e) {
                System.out.println("No matching element found");
            } catch (NonIntResultException e) {
                System.out.println(e);
            }
        }
    }
}
class NonIntResultException extends Exception {
    int n, d;

    NonIntResultException(int i, int j) {
        n = i;
        d = j;
    }

    public String toString() {
        return "Result of " + n + " / " + d + " is non integer";
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\CustomExceptionDemo.java

4 / 2 is 2

Cannot divide by Zero

Result of 15 / 4 is non integer

32 / 4 is 8

Q. Add exception to queue

```java
class QueueFullException extends Exception {
    int size;

    QueueFullException(int s) {
        size = s;
    }

    public String toString() {
        return "\nQueue is full. Maximum size is " + size;
    }
}

class QueueEmptyException extends Exception {
    QueueEmptyException() {
    }

    public String toString() {
        return "\nQueue is Empty.";
    }
}

class Queue {
    private char[] q;
    private int putloc, getloc;

    Queue(int size) {
        q = new char[size];
        putloc = getloc = 0;
    }

    void put(char ch) throws QueueFullException {
        if (putloc == q.length) {
            throw new QueueFullException(q.length);
        }
        q[putloc++] = ch;
    }

    char get() throws QueueEmptyException {
        if (getloc == putloc)
            throw new QueueEmptyException();

        return q[getloc++];
    }
}

public class QueueDemo {
```

```java
    public static void main(String[] args) throws Exception {
        Queue bigQ = new Queue(100);
        Queue smallQ = new Queue(4);
        char ch;
        int i;

        for (i = 0; i < 26; i++)
            bigQ.put((char) ('A' + i));

        for (i = 0; i < 26; i++) {
            ch = bigQ.get();
            if (ch != (char) 0)
                System.out.print(ch);
        }

        System.out.println("\n");

        System.out.println("Using smallQ to generate errors.");
        for (i = 0; i < 5; i++) {
            System.out.println("Attempting to store " + (char) ('Z' - i));

            smallQ.put((char) ('Z' - i));

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

        System.out.println("Contents of SmallQ: ");
        for (i = 0; i < 5; i++) {
            ch = smallQ.get();
            if (ch != (char) 0)
                System.out.print(ch);
        }
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\QueueDemo.java

ABCDEFGHIJKLMNOPQRSTUVWXYZ

Using smallQ to generate errors.

Attempting to store Z

Attempting to store Y

Attempting to store X

Attempting to store W

Attempting to store V

Exception in thread "main"

Queue is full. Maximum size is 4

        at Queue.put(QueueDemo.java:33)

        at QueueDemo.main(QueueDemo.java:68)

Q. Try It Out

```java
class ZeroDivideException extends Exception {
    private int index = -1;

    public ZeroDivideException() {
    }

    public ZeroDivideException(String s) {
        super(s);
    }

    public ZeroDivideException(int index) {
        super("/ by zero");
        this.index = index;
    }

    public int getIndex() {
        return index;
    }
}

public class TryItOut {
    public static int divide(int[] array, int index) throws ZeroDivideException {
        try {
            System.out.println("First try block in divide() entered");
            array[index + 2] = array[index] / array[index + 1];
            System.out.println("Code at end of first try block in divide()");
            return array[index + 2];
        } catch (ArithmeticException e) {
            System.out.println("Arithmetic exception caught in divide()");
            throw new ZeroDivideException(index + 1);
        } catch (ArrayIndexOutOfBoundsException e) {
            System.out.println("Index-out-of-
bounds index exception caught in divide()");
        }
        System.out.println("Executing code after try block in divide()");
        return array[index + 2];
    }

    public static void main(String[] args) {
        int[] x = { 10, 5, 0 };
        try {
            System.out.println("First try block in main()entered");
            System.out.println("result = " + divide(x, 0));
            x[1] = 0;
            System.out.println("result = " + divide(x, 0));
```

```java
            x[1] = 1;
            System.out.println("result = " + divide(x, 1));
        } catch (ZeroDivideException e) {
            int index = e.getIndex();
            if (index > 0) {
                x[index] = 1;
                x[index + 1] = x[index - 1];
                System.out.println("Zero divisor corrected to " + x[index]);
            }
        } catch (ArithmeticException e) {
            System.out.println("Arithmetic exception caught in main()");
        } catch (ArrayIndexOutOfBoundsException e) {
            System.out.println("Index-out-of-
bounds exception caught in main()");
        }
        System.out.println("Outside first try block in main()");
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\TryItOut.java

First try block in main()entered

First try block in divide() entered

Code at end of first try block in divide()

result = 2

First try block in divide() entered

Arithmetic exception caught in divide()

Zero divisor corrected to 1

Outside first try block in main()

Q. Java 368

```java
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.IOException;

public class ShowFile {
    public static void main(String[] args) {

        int i;
        FileInputStream fin;

        if (args.length != 1) {
            System.out.println("Usgae: ShowFile File");
            return;
        }

        try {
            fin = new FileInputStream(args[0]);
        } catch (FileNotFoundException e) {
            System.out.println(e.getMessage());
            return;
        }

        try {
            do {
                i = fin.read();
                if (i != -1)
                    System.out.print((char) i);
            } while (i != -1);
        } catch (IOException e) {
            System.out.println(e.getMessage());
        }

        try {
            fin.close();
        } catch (IOException e) {
            System.out.println(e.getMessage());
        }
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\ShowFile.java new.txt

 Some data in file

Q. Help -> 10.2

```java
import java.io.*;

class Help {
    String helpfile;

    Help(String fname) {
        helpfile = fname;
    }

    boolean helpOn(String what) {
        int ch;
        String topic, info;

        try (BufferedReader helprdr = new BufferedReader(new FileReader(helpfi
le))) {
            do {
                ch = helprdr.read();
                if (ch == '#') {
                    topic = helprdr.readLine();
                    if (what.compareTo(topic) == 0) {
                        do {
                            info = helprdr.readLine();
                            if (info != null)
                                System.out.println(info);
                        } while ((info != null) && (info.compareTo("") != 0));
                        return true;
                    }
                }
            } while (ch != 1);
        } catch (IOException e) {
            System.out.println(e.getMessage());
            return false;
        }
        return false;
    }

    String getSelection() {
        String topic = "";
        BufferedReader br = new BufferedReader(new InputStreamReader(System.in
));

        System.out.print("Enter topic: ");
        try {
            topic = br.readLine();
        } catch (IOException e) {
            System.out.println(e.getMessage());
        }
```

```java
        return topic;
    }
}

public class FileHelp {
    public static void main(String[] args) {
        Help hlpobj = new Help("helpfile.txt");
        String topic;

        System.out.println("try the help system." + "Enter 'stop' to end");

        do {
            topic = hlpobj.getSelection();

            if (!hlpobj.helpOn(topic))
                System.out.println("Topic not found");

        } while (topic.compareTo("stop") != 0);
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\FileHelp.java

try the help system.Enter 'stop' to end

Enter topic: topic2

this is help on topic 2

Enter topic: topic1

this is help on topic 1

Q. Write A String

```java
import java.io.File;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.FileNotFoundException;
import java.nio.ByteBuffer;
import java.nio.channels.FileChannel;

public class WriteAString {
    public static void main(String[] args) {
        String phrase = new String("Garbage in, garbage out\n");
        String dirname = "D:/New";
        String filename = "charData.txt";
        File dir = new File(dirname);

        if (!dir.exists()) {
            if (!dir.mkdir()) {
                System.out.println("Cannot create directory: " + dirname);
                System.exit(1);
            }
        } else if (!dir.isDirectory()) {
            System.err.println(dirname + " is not a directory");
            System.exit(1);
        }
        File aFile = new File(dir, filename);
        FileOutputStream outputFile = null;
        try {
            outputFile = new FileOutputStream(aFile, true);
            System.out.println("File stream created successfully.");
        } catch (FileNotFoundException e) {
            e.printStackTrace(System.err);
        }

        FileChannel outChannel = outputFile.getChannel();
        ByteBuffer buf = ByteBuffer.allocate(1024);
        System.out.println("New buffer: position = " + buf.position() + "\tLim
it = " + buf.limit() + "\tcapacity = "
                + buf.capacity());

        for (char ch : phrase.toCharArray()) {
            buf.putChar(ch);
        }
        System.out.println("Buffer after loading: position = " + buf.position(
) + "\tLimit = " + buf.limit()
                + "\tcapacity = " + buf.capacity());
        buf.flip();
```

```
        System.out.println("Buffer after flip: position = " + buf.position() +
"\tLimit = " + buf.limit()
                + "\tcapacity = " + buf.capacity());

        try {
            outChannel.write(buf);
            outputFile.close();
            System.out.println("Buffer contents written to file.");
        } catch (IOException e) {
            e.printStackTrace(System.err);
        }
        System.exit(0);
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\WriteAString.java

File stream created successfully.

New buffer: position = 0   Limit = 1024   capacity = 1024

Buffer after loading: position = 48   Limit = 1024   capacity = 1024

Buffer after flip: position = 0 Limit = 48   capacity = 1024

Buffer contents written to file.

Q. Write A String As Byte

```java
import java.io.File;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.FileNotFoundException;
import java.nio.ByteBuffer;
import java.nio.channels.FileChannel;

public class WriteAStringAsBytes {
    public static void main(String[] args) {
        String phrase = new String("Garbage in, garbage out\n");
        String dirname = "D:/New";
        String filename = "byteData.txt";
        File aFile = new File(dirname, filename);

        FileOutputStream file = null;
        try {
            file = new FileOutputStream(aFile, true);
        } catch (FileNotFoundException e) {
            e.printStackTrace(System.err);
        }
        FileChannel outChannel = file.getChannel();
        ByteBuffer buf = ByteBuffer.allocate(phrase.length());
        byte[] bytes = phrase.getBytes();
        buf.put(bytes);
        buf.flip();
        try {
            outChannel.write(buf);
            file.close();
        } catch (IOException e) {
            e.printStackTrace(System.err);
        }
    }
}
```

Run java file :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\WriteAStringAsByte.java

Q. Proverbs

```java
import java.io.File;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.FileNotFoundException;
import java.nio.ByteBuffer;
import java.nio.channels.FileChannel;

public class WriteProverbs {
    public static void main(String[] args) {
        String dirName = "D:/New";
        String fileName = "Proverbs.txt";
        String[] sayings = { "Indecision maximizes flexibility.", "Only the mediocre are always at their best.",
                "A little knowledge is a dangerous thing.", "Many a mickle makes a muckle.",
                "Who begins too much achieves little.", "Who knows most says least.",
                "A wise man sits on the hole in his carpet." };
        File aFile = new File(dirName, fileName);
        FileOutputStream outputFile = null;
        try {
            outputFile = new FileOutputStream(aFile, true);
        } catch (FileNotFoundException e) {

            e.printStackTrace(System.err);
            System.exit(1);
        }
        FileChannel outChannel = outputFile.getChannel();

        int maxLength = 0;
        for (String saying : sayings) {
            if (maxLength < saying.length())
                maxLength = saying.length();
        }
        ByteBuffer buf = ByteBuffer.allocate(2 * maxLength + 4);

        try {
            for (String saying : sayings) {
                buf.putInt(saying.length()).asCharBuffer().put(saying);
                buf.position(buf.position() + 2 * saying.length()).flip();
                outChannel.write(buf);
                buf.clear();
            }
            outputFile.close();
            System.out.println("Proverbs written to file.");
        } catch (IOException e) {
```

```
            e.printStackTrace(System.err);
            System.exit(1);
        }
        System.exit(0);
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\WriteProverbs.java

Proverbs written to file

Q. Use A Formatter

```java
import java.io.File;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.FileNotFoundException;
import java.nio.ByteBuffer;
import java.nio.CharBuffer;
import java.nio.channels.FileChannel;
import java.util.Formatter;

public class UsingAFormatter {
    public static void main(String[] args) {
        String[] phrases = { "Rome wasn't burned in a day.", "It's a bold mouse that sits in the cat's ear.",
                "An ounce of practice is worth a pound of instruction." };
        String dirname = "D:/New";
        String filename = "Phrases.txt";
        File dir = new File(dirname);

        if (!dir.exists()) {
            if (!dir.mkdir()) {
                System.out.println("Cannot create directory: " + dirname);
                System.exit(1);
            }
        } else if (!dir.isDirectory()) {
            System.err.println(dirname + " is not a directory");
            System.exit(1);
        }

        File aFile = new File(dir, filename);
        FileOutputStream outputFile = null;
        try {
            outputFile = new FileOutputStream(aFile, true);
            System.out.println("File stream created successfully.");
        } catch (FileNotFoundException e) {
            e.printStackTrace(System.err);
        }

        FileChannel outChannel = outputFile.getChannel();

        ByteBuffer buf = ByteBuffer.allocate(1024);
        System.out.println("\nByte buffer:");
        System.out.printf("position = %2d Limit = %4d capacity = %4d%n", buf.position(), buf.limit(), buf.capacity());

        CharBuffer charBuf = buf.asCharBuffer();
```

```java
        System.out.println("Char view buffer:");
        System.out.printf("position = %2d Limit = %4d capacity = %4d%n", charB
uf.position(), charBuf.limit(),
                charBuf.capacity());
        Formatter formatter = new Formatter(charBuf);

        int number = 0;
        for (String phrase : phrases) {
            formatter.format("%nProverb%3d: %s", ++number, phrase);
            System.out.println("\nView buffer after loading:");
            System.out.printf("position = %2d Limit = %4d capacity = %4d%n", c
harBuf.position(), charBuf.limit(),
                    charBuf.capacity());
            charBuf.flip();
            System.out.println("View buffer after flip:");
            System.out.printf("position = %2d Limit = %4d length = %4d%n", cha
rBuf.position(), charBuf.limit(),
                    charBuf.length());
            buf.limit(2 * charBuf.length());
            System.out.println("Byte buffer after limit update:");
            System.out.printf("position = %2d Limit = %4d length = %4d%n", buf
.position(), buf.limit(),
                    buf.remaining());

            try {
                outChannel.write(buf);
                System.out.println("Buffer contents written to file.");
                buf.clear();
                charBuf.clear();
            } catch (IOException e) {
                e.printStackTrace(System.err);
                System.exit(1);
            }
        }
        try {
            outputFile.close();
        } catch (IOException e) {
            e.printStackTrace(System.err);
        }
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\UsingAFormatter.java
File stream created successfully.

Byte buffer:
position = 0 Limit = 1024 capacity = 1024
Char view buffer:
position = 0 Limit = 512 capacity = 512

View buffer after loading:
position = 44 Limit = 512 capacity = 512
View buffer after flip:
position = 0 Limit = 44 length = 44
Byte buffer after limit update:
position = 0 Limit = 88 length = 88
Buffer contents written to file.

View buffer after loading:
position = 63 Limit = 512 capacity = 512
View buffer after flip:
position = 0 Limit = 63 length = 63
Byte buffer after limit update:
position = 0 Limit = 126 length = 126
Buffer contents written to file.

View buffer after loading:
position = 67 Limit = 512 capacity = 512
View buffer after flip:
position = 0 Limit = 67 length = 67
Byte buffer after limit update:
position = 0 Limit = 134 length = 134
Buffer contents written to file.

Q. Try It Out Copying File

```java
import java.io.File;
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.FileNotFoundException;
import java.io.IOException;
import java.nio.channels.FileChannel;

public class FileCopy {
    public static void main(String[] args) {
        if (args.length == 0) {
            System.out.println("No file to copy. Application usage is:\n" + "j
ava -classpath . FileCopy \"filepath\"");
            System.exit(1);
        }
        File fromFile = new File(args[0]);
        if (!fromFile.exists()) {
            System.out.printf("File to copy, %s, does not exist.", fromFile.ge
tAbsolutePath());
            System.exit(1);
        }
        File toFile = createBackupFile(fromFile);
        FileInputStream inFile = null;
        FileOutputStream outFile = null;
        try {
            inFile = new FileInputStream(fromFile);
            outFile = new FileOutputStream(toFile);
        } catch (FileNotFoundException e) {
            e.printStackTrace(System.err);
            assert false;
        }
        FileChannel inChannel = inFile.getChannel();
        FileChannel outChannel = outFile.getChannel();
        try {
            int bytesWritten = 0;
            long byteCount = inChannel.size();
            while (bytesWritten < byteCount) {
                bytesWritten += inChannel.transferTo(bytesWritten, byteCount -
 bytesWritten, outChannel);
            }
            System.out.printf("File copy complete. %d bytes copied to %s%n", b
yteCount, toFile.getAbsolutePath());
            inFile.close();
            outFile.close();
        } catch (IOException e) {
            e.printStackTrace(System.err);
            System.exit(1);
```

```java
        }
        System.exit(0);
    }

    public static File createBackupFile(File aFile) {
        aFile = aFile.getAbsoluteFile();
        File parentDir = new File(aFile.getParent());
        String name = aFile.getName();
        int period = name.indexOf('.');
        if (period == -1) {
            period = name.length();
        }
        String nameAdd = "_backup";
        File backup = aFile;
        while (backup.exists()) {
            name = backup.getName();
            backup = new File(parentDir, name.substring(0, period) + nameAdd +
 name.substring(period));
            period += nameAdd.length();
        }
        return backup;
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Practice> java .\FileCopy.java byteData.txt
File copy complete. 12 bytes copied to D:\MCA\MCA SEM
3\JAVA\Practice\byteData_backup.txt

Q. Read & Write

```java
import java.io.File;
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.FileNotFoundException;
import java.io.IOException;
import java.nio.ByteBuffer;
import java.nio.channels.FileChannel;

public class RandomReadWrite {
    public static void main(String[] args) {
        File aFile = new File("primes.txt");
        FileInputStream inFile = null;
        FileOutputStream outFile = null;
        try {
            inFile = new FileInputStream(aFile);
            outFile = new FileOutputStream(aFile, true);
        } catch (FileNotFoundException e) {
            e.printStackTrace(System.err);
            System.exit(1);
        }
        FileChannel inChannel = inFile.getChannel();
        FileChannel outChannel = outFile.getChannel();
        final int PRIMESREQUIRED = 10;
        ByteBuffer buf = ByteBuffer.allocate(8);
        long[] primes = new long[PRIMESREQUIRED];
        int index = 0;
        final long REPLACEMENT = 99999L;
        try {
            final int PRIMECOUNT = (int) inChannel.size() / 8;
            System.out.println("Prime count = " + PRIMECOUNT);
            for (int i = 0; i < PRIMESREQUIRED; i++) {
                index = 8 * (int) (PRIMECOUNT * Math.random());
                inChannel.read(buf, index);
                buf.flip();
                primes[i] = buf.getLong();
                buf.flip();
                buf.putLong(REPLACEMENT);
                buf.flip();
                outChannel.write(buf, index);
                buf.clear();
            }
            int count = 0;
            for (long prime : primes) {
                System.out.printf("%12d", prime);
                if (++count % 5 == 0) {
                    System.out.println();
```

```
                }
            }
            inFile.close();
            outFile.close();
        } catch (IOException e) {
            e.printStackTrace(System.err);
            System.exit(1);
        }
        System.exit(0);
    }
}
```

Output :

359   107   383   109   7
173   443   337   17   113

Q. Use Threads

```java
class UseThreads {
    public static void main(String[] args) {
        System.out.println("Main thread starting.");

        MyThread mt = new MyThread("Child #1");

        Thread newThrd = new Thread(mt);

        newThrd.start();

        for(int i=0;i<50;i++)
        {
            System.out.println(".\n" + newThrd.isAlive());
            System.out.println(".");
            try{
                Thread.sleep(200);
            }
            catch(InterruptedException exc){
                System.out.println("Main thread interrupted.");
            }
            newThrd.interrupt();
        }
        System.out.println("Main thread ending.");
    }
}

class MyThread implements Runnable{
    String thrdName;
    MyThread(String name) {
        thrdName = name;
    }

    public void run() {
        System.out.println(thrdName + "starting.");
        try {
            for(int count=0;count<10;count++) {
                Thread.sleep(5000);
                System.out.println("In " + thrdName + ", count is " +count);
            }
        }
        catch(InterruptedException exc) {
            System.out.println(thrdName + " interrupted.");
        }

        System.out.println(thrdName + " terminating.");
    }
}
```

```
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA\Thread> java .\UseThreads.java
Main thread starting.
.
true
.
Child #1starting.
.
true
.
Child #1 interrupted.
Child #1 terminating.
.
false
.
.
false
.
.
false
.
.
false
.
.
false
.
.
false
.
.
false
.
.
false
.
.
false

-
-
false
-
-
false
-
-
false
-
-
false
-
-
false
-
-
false
-
-
false
-
-
false
-
-
false
-
-
false
-
-
false
-
-
false
-

- .
- **false**
- .
- .
- **false**
- .
- .
- **false**
- .
- .
- **false**
- .
- .
- **false**
- .
- .
- **false**
- .
- .
- **false**
- .
- .
- **false**
- .
- .
- **false**
- .
- .
- **false**
- .
- .
- **false**
- .
- .
- **false**
- .
- .
- **false**
- .
- .

false

.

.

false

.

.

false

.

.

false

.

.

false

.

.

false

.

.

false

.

.

false

.

.

false

.

.

false

.

.

false

.

.

false

.

Main thread ending.

Q. Sum Array Synchronization

```java
class SumArray {
    private int sum;

    synchronized int sumArray(int[] nums) {
        sum = 0;

        for (int i = 0; i < nums.length; i++) {
            sum += nums[i];
            System.out.println("Running total for " + Thread.currentThread().getName() + " is " + sum);

            try {
                Thread.sleep(10);
            } catch (InterruptedException exc) {
                System.out.println("Thread interrupted!");
            }
        }
        return sum;
    }
}

class MyThread implements Runnable {
    Thread thrd;
    static SumArray sa = new SumArray();
    int[] a;
    int answer;

    MyThread(String name, int[] nums) {
        thrd = new Thread(this, name);
        a = nums;
        thrd.start();
    }

    public void run() {
        int sum;

        System.out.println(thrd.getName() + " starting...");

        answer = sa.sumArray(a);
        System.out.println("Sum for " + thrd.getName() + " is " + answer);

        System.out.println(thrd.getName() + " terminating...");
    }
}

public class SumArraySync {
```

```java
    public static void main(String[] args) {
        int[] a = { 1, 2, 3, 4, 5 };

        MyThread mt1 = new MyThread("Child #1", a);
        MyThread mt2 = new MyThread("Child #2", a);

        try {
            mt1.thrd.join();
            mt2.thrd.join();
        } catch (InterruptedException e) {
            System.out.println("Main thread interrupted!");
        }
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA> java .\SumArraySync.java
Child #2 starting...
Child #1 starting...
Running total for Child #2 is 1
Running total for Child #2 is 3
Running total for Child #2 is 6
Running total for Child #2 is 10
Running total for Child #2 is 15
Running total for Child #1 is 1
Sum for Child #2 is 15
Child #2 terminating...
Running total for Child #1 is 3
Running total for Child #1 is 6
Running total for Child #1 is 10
Running total for Child #1 is 15
Sum for Child #1 is 15
Child #1 terminating...

Q. Class TikTok

```java
public class ThreadCom {
    public static void main(String[] args) {
        TickTock t1 = new TickTock();
        MyThread mt1 = new MyThread("Tick", t1);
        MyThread mt2 = new MyThread("Tock", t1);

        try {
            mt1.thrd.join();
            mt2.thrd.join();
        } catch (InterruptedException e) {
            System.out.println("Main thread interrupted...");
        }
    }
}

class TickTock {
    String state;

    synchronized void tick(boolean running) {
        if (!running) {
            state = "ticked";
            notify();
            return;
        }

        System.out.println("Tick ");

        state = "ticked";
        notify();

        try {
            while (!state.equals("tocked"))
                wait();
        } catch (InterruptedException e) {
            System.out.println("Thread interrupted");
        }
    }

    synchronized void tock(boolean running) {
        if (!running) {
            state = "tocked";
            notify();
            return;
        }

        System.out.println("Tock ");
```

```java
        state = "tocked";
        notify();

        try {
            while (!state.equals("ticked"))
                wait();
        } catch (InterruptedException e) {
            System.out.println("Thread interrupted");
        }
    }
}

class MyThread implements Runnable {
    Thread thrd;
    TickTock ttOb;

    MyThread(String name, TickTock tt) {
        thrd = new Thread(this, name);
        ttOb = tt;
        thrd.start();
    }

    public void run() {
        if (thrd.getName().compareTo("Tick") == 0) {
            for (int i = 0; i < 5; i++)
                ttOb.tick(true);
            ttOb.tick(false);
        } else {
            for (int i = 0; i < 5; i++)
                ttOb.tock(true);
            ttOb.tock(false);
        }
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA> java .\ThreadCom.java
Tick
Tock
Tick
Tock
Tick

Tock
Tick
Tock
Tick
Tock

Q. Suspend.java

```java
class Suspend {
    public static void main(String[] args) {
        MyThread ob1 = new MyThread("My Thread");
        try {
            Thread.sleep(1000);

            ob1.mysuspend();
            System.out.println("Suspending thread.");
            Thread.sleep(1000);

            ob1.myresume();
            System.out.println("Resuming thread.");
            Thread.sleep(1000);

            ob1.mysuspend();
            System.out.println("Suspending thread.");
            Thread.sleep(1000);

            ob1.myresume();
            System.out.println("Resuming thread.");
            Thread.sleep(1000);

            ob1.mysuspend();
            System.out.println("Stopping thread.");
            ob1.mystop();
        } catch (InterruptedException e) {
            System.out.println("Main thread interrupted.");

        }
    }
}

class MyThread implements Runnable {
    Thread thrd;

    boolean suspended;
    boolean stopped;

    MyThread(String name) {
        thrd = new Thread(this, name);
        suspended = false;
        stopped = false;
        thrd.start();
    }

    // this is the entry point for thread
```

```java
    public void run() {
        System.out.println(thrd.getName() + " Starting.");
        try {
            for (int i = 1; i < 1000; i++) {
                System.out.println(i + " ");
                if ((i % 10) == 0) {
                    System.out.println();
                    Thread.sleep(250);
                }
                synchronized (this) {
                    while (suspended) {
                        wait();
                    }
                    if (stopped)
                        break;
                }
            }
        } catch (InterruptedException e) {
            System.out.println(thrd.getName() + " Interrupted.");
        }
        System.out.println(thrd.getName() + " exiting.");

    }

    synchronized void mystop() {
        stopped = true;
        suspended = false;
        notify();
    }

    synchronized void mysuspend() {
        suspended = true;
    }

    synchronized void myresume() {
        suspended = false;
        notify();
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA> java .\Suspend.java
My Thread Starting.
1
2

3
4
5
6
7
8
9
10

11
12
13
14
15
16
17
18
19
20

21
22
23
24
25
26
27
28
29
30

31
32
33
34
35
36
37
38
39

40

Suspending thread.
Resuming thread.
41
42
43
44
45
46
47
48
49
50

51
52
53
54
55
56
57
58
59
60

61
62
63
64
65
66
67
68
69
70

71
72
73

74
75
76
77
78
79
80

Suspending thread.
Resuming thread.
81
82
83
84
85
86
87
88
89
90

91
92
93
94
95
96
97
98
99
100

101
102
103
104
105
106
107
108

109
110

Stopping thread.
My Thread exiting.

Exercise - 3

Q. Write a program that, using an integer array of date values containing month, day, and year as integers for some number of dates (10, say, so the integer array will be two-dimensional with 10 rows and 3 columns), will write a file with a string representation of each date written as Unicode characters. For example, the date values 3,2,1990 would be written to the file as 2nd March 1990. Make sure that the date strings can be read back, either by using a separator character of some kind to mark the end of each string or by writing the length of each string before you write the string itself.

```java
import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.FileReader;
import java.io.FileWriter;
import java.io.IOException;

public class DateReadWrite {
    public static void main(String[] args) throws IOException {
        int[][] dates = { { 21, 1, 1998 }, { 1, 2, 1999 }, { 12, 3, 2000 }, {
26, 4, 2001 }, { 13, 5, 2002 },
                { 2, 6, 2003 }, { 5, 7, 2004 }, { 10, 8, 2005 }, { 20, 9, 2006
}, { 31, 10, 2007 }, };

        FileWriter writer = new FileWriter("dates.txt");

        for (int i = 0; i < dates.length; i++) {
            Date d = new Date(dates[i][0], dates[i][1], dates[i][2]);
            String date = d.toString();
            System.out.println("Written: " + date);
            writer.write(date + "\n");
        }

        System.out.println("\nReading Data: ");
        writer.close();

        BufferedReader reader = new BufferedReader(new FileReader("dates.txt")
);
        String date;
```

```java
        while ((date = reader.readLine()) != null) {
            System.out.println("Read: " + Date.getDate(date));
        }

        reader.close();
    }
}

class Date {
    private static final String[] months = { "January", "February", "March", "
April", "May", "June", "July", "August",
            "September", "October", "November", "December" };

    int dd, mm, yy;

    Date(int dd, int mm, int yy) {
        this.dd = dd;
        this.mm = mm;
        this.yy = yy;
    }

    private static String getDate(int dd) {
        String suffix = "th";
        if (dd == 1 || dd == 21 || dd == 31)
            suffix = "st";
        else if (dd == 2 || dd == 22)
            suffix = "nd";
        else if (dd == 3 || dd == 23)
            suffix = "rd";
        return dd + suffix;
    }

    private String getMonth(int mm) {
        return months[mm - 1];
    }

    private static int getMonth(String month) {
        switch (month) {
            case "January":
                return 1;
            case "February":
                return 2;
            case "March":
                return 3;
            case "April":
                return 4;
            case "May":
                return 5;
```

```java
            case "June":
                return 6;
            case "July":
                return 7;
            case "August":
                return 8;
            case "September":
                return 9;
            case "October":
                return 10;
            case "November":
                return 11;
            default:
                return 12;
        }
    }

    static Date getDate(String date) {
        String[] values = date.split("[-]");
        int dd = Integer.parseInt(values[0].substring(0, values[0].length() -
2));
        int mm = getMonth(values[1]);
        int yy = Integer.parseInt(values[2]);
        return new Date(dd, mm, yy);
    }

    public String toString() {
        return (getDate(this.dd) + "-" + getMonth(this.mm) + "-" + this.yy);
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA> java .\DateReadWrite.java
Written: 21st-January-1998
Written: 1st-February-1999
Written: 12th-March-2000
Written: 26th-April-2001
Written: 13th-May-2002
Written: 2nd-June-2003
Written: 5th-July-2004
Written: 10th-August-2005
Written: 20th-September-2006

Written: 31st-October-2007

Reading Data:
Read: 21st-January-1998
Read: 1st-February-1999
Read: 12th-March-2000
Read: 26th-April-2001
Read: 13th-May-2002
Read: 2nd-June-2003
Read: 5th-July-2004
Read: 10th-August-2005
Read: 20th-September-2006
Read: 31st-October-2007

Q. Extend the previous example to write a second file at the same time as the first, but containing the month, day, and year values as binary data. You should have both files open and be writing to both at the same time.

```java
import java.io.*;

class SyncDateReadWrite {
    public static void main(String[] args) throws IOException {
        int[][] dates = { { 21, 1, 1998 }, { 1, 2, 1999 }, { 12, 3, 2000 }, {
26, 4, 2001 }, { 13, 5, 2002 },
                    { 2, 6, 2003 }, { 5, 7, 2004 }, { 10, 8, 2005 }, { 20, 9, 2006
 }, { 31, 10, 2007 }, };

        FileWriter writer = new FileWriter("dates.txt");
        DataOutputStream os = new DataOutputStream(new FileOutputStream("binar
y.dat"));

        for (int i = 0; i < dates.length; i++) {
            Date d = new Date(dates[i][0], dates[i][1], dates[i][2]);
            os.writeInt(dates[i][0]);
            os.writeInt(dates[i][1]);
            os.writeInt(dates[i][2]);
            String date = d.toString();
            System.out.println("Written: " + date);
            writer.write(date + "\n");
        }

        System.out.println("\nReading Data: ");
        writer.close();
        os.close();

        BufferedReader reader = new BufferedReader(new FileReader("dates.txt")
);
        DataInputStream is = new DataInputStream(new FileInputStream("binary.d
at"));
        String date;

        while ((date = reader.readLine()) != null) {
            System.out.println("Read: " + Date.getDate(date));
        }
        System.out.println("\nReading binary file: \n ");

        int datesRead[][] = new int[20][];
        int j = 0;
        while (is.available() > 0) {
            int dd = is.readInt();
            int mm = is.readInt();
            int yy = is.readInt();
```

```java
            datesRead[j] = new int[3];
            datesRead[j][0] = dd;
            datesRead[j][1] = mm;
            datesRead[j][2] = yy;
            j++;

            System.out.print(dd + " - ");
            System.out.print(mm + " - ");
            System.out.println(yy);
        }

        reader.close();
        is.close();
    }
}

class Date {
    private static final String[] months = { "January", "February", "March", "
April", "May", "June", "July", "August",
            "September", "October", "November", "December" };

    int dd, mm, yy;

    Date(int dd, int mm, int yy) {
        this.dd = dd;
        this.mm = mm;
        this.yy = yy;
    }

    private static String getDate(int dd) {
        String suffix = "th";
        if (dd == 1 || dd == 21 || dd == 31)
            suffix = "st";
        else if (dd == 2 || dd == 22)
            suffix = "nd";
        else if (dd == 3 || dd == 23)
            suffix = "rd";
        return dd + suffix;
    }

    private String getMonth(int mm) {
        return months[mm - 1];
    }

    private static int getMonth(String month) {
        switch (month) {
            case "January":
                return 1;
```

```java
            case "February":
                return 2;
            case "March":
                return 3;
            case "April":
                return 4;
            case "May":
                return 5;
            case "June":
                return 6;
            case "July":
                return 7;
            case "August":
                return 8;
            case "September":
                return 9;
            case "October":
                return 10;
            case "November":
                return 11;
            default:
                return 12;
        }
    }

    static Date getDate(String date) {
        String[] values = date.split("[-]");
        int dd = Integer.parseInt(values[0].substring(0, values[0].length() -
2));
        int mm = getMonth(values[1]);
        int yy = Integer.parseInt(values[2]);
        return new Date(dd, mm, yy);
    }

    public String toString() {
        return (getDate(this.dd) + "-" + getMonth(this.mm) + "-" + this.yy);
    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA> java .\SyncDateReadWrite.java
Written: 21st-January-1998
Written: 1st-February-1999
Written: 12th-March-2000

Written: 26th-April-2001
Written: 13th-May-2002
Written: 2nd-June-2003
Written: 5th-July-2004
Written: 10th-August-2005
Written: 20th-September-2006
Written: 31st-October-2007

Reading Data:
Read: 21st-January-1998
Read: 1st-February-1999
Read: 12th-March-2000
Read: 26th-April-2001
Read: 13th-May-2002
Read: 2nd-June-2003
Read: 5th-July-2004
Read: 10th-August-2005
Read: 20th-September-2006
Read: 31st-October-2007

Reading binary file:

21 - 1 - 1998
1 - 2 - 1999
12 - 3 - 2000
26 - 4 - 2001
13 - 5 - 2002
2 - 6 - 2003
5 - 7 - 2004
10 - 8 - 2005
20 - 9 - 2006
31 - 10 – 2007

Q. Write a program that, for a given String object defined in the code, will write strings to a file in the local character encoding (as bytes) corresponding to all possible permutations of the words in the string. For example, for the string the fat cat, you would write the strings the fat
cat, the cat fat, cat the fat, cat fat the, fat the cat, and fat cat the, to the file, although not necessarily in that sequence. (Don't use very long strings; with n words in the string, the number of permutations is n!).

```java
import java.io.*;

public class Permutation {
    public static void main(String[] args) throws IOException {
        String str = "the fat cat";
        String[] words = str.split("[ ]");

        FileWriter writer = new FileWriter("permutations.txt");

        for (int i = 0; i < 3; i++) {
            for (int j = 0; j < 3; j++) {
                for (int k = 0; k < 3; k++) {
                    String data = words[i] + words[j] + words[k] + "\n";
                    writer.write(data);
                }
            }
        }

        writer.close();

        BufferedReader reader = new BufferedReader(new FileReader("permutations.txt"));

        while ((str = reader.readLine()) != null)
            System.out.println(str);

        reader.close();

    }
}
```

Output :

PS D:\MCA\MCA SEM 3\JAVA> java .\Permutation.java

thethethe

thethefat

thethecat

thefatthe

thefatfat

thefatcat

thecatthe

thecatfat

thecatcat

fatthethe

fatthefat

fatthecat

fatfatthe

fatfatfat

fatfatcat

fatcatthe

fatcatfat

fatcatcat

catthethe

catthefat

catthecat

catfatthe

catfatfat

catfatcat

catcatthe

catcatfat

catcatcat