PRACTICAL PROGRAMS

Name : Pradip S Karmakar

Roll No : 10

Class : M.C.A – 3

Subject : Java Programming

Q1 . 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!

Q2 . Write a program using Command Line Arguments.

```java
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

Q3 . 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

Q4 . 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

Q5 . 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)
            {
```

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

Output :

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.

Q6 . 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

Q7 . 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

Q8 . 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

Q9 . 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.

Q10 . 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

Q11 . 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

Q12 . 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

Q13 . 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

Q14 . 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

Q15 . 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

Q16 . 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(pack
ed));

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

Q17 . 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));
    }
}
```

<u>Output :</u>

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

Q18 . 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 :
11111110000000000000000000000000000000000000000000000000000000
0001

Previous result reversed :
100000000000000000000000000000000000000000000000000000000000111
1111

Bit count in number : 8

Q19 . 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 " + symbol);
            }
            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

Q20 . 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

Q21 . 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.

Q22 . 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

Q23 . Demonstrate Collection-Based for loop.

```java
public class CollectionForLoop {
    enum Season { spring, summer, monsoon, winter } // Enumeration typ
e 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

Q24 . 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

Q25 . 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

Q26 . 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

Q27 . 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

Q28 . 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 exac
t division

                    break; // so exit the loop
                }
            }
            // We can get here through the break, or through completin
g 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
        }
    }
}
```

Output :

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

Q29. 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

Q30. 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");

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

```java
            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':
            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

Q31. 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 " + sroo
t);

            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

Q32. 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

Q33. 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

Q34. 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

Q35. 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 ****

Q36. 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