

Q.1.

Write a program to print numbers from 1 to 10.

```
class Numbers
```

```
{
```

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

```
for (int i = 1; i <= 10; i++)
{
```

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

```
}
```

Output:- 1 2 3 4 5 6 7 8 9 10

1

2

3

4

5

6

7

8

9

10

Q.2. Write a program to calculate the sum of first 10 natural numbers.

```
public class Demo  
{  
    public static void main(String[] args)  
    {  
        int num = 10, count = 1, total = 0;  
        while (count <= num)  
        {  
            total = total + count;  
            count++;  
        }  
        System.out.println("Sum of first 10  
natural numbers is :" + total);  
    }  
}
```

Output:-

Sum of first 10 natural numbers is
55

Q.3. Write a program that prompts the user to input a positive integer. It should then print the multiplication table of that number.

```
class MultiplicationTable
{
    public static void main(String args[])
    {
        int num = Integer.parseInt(args[0]);
        System.out.println("multiplication Table
                           of " + num);
        for (int j = 1; j <= 10; j++)
        {
            System.out.print(" " + num * j + " ");
        }
        System.out.print("\n");
    }
}
```

Output:-

6 12 18 24 30 36

12 24 36

18 36

24

30

36

Q.4. Write a program to find the factorial value of any number entered through the keyboard.

```
public class Factorial
{
    public static void main(String args[])
    {
        int num = 10;
        long factorial = 1;
        for (int i = 1; i <= num; i++)
        {
            factorial *= i;
        }
        System.out.println("Factorial of " + num + " is " + factorial);
    }
}
```

Output :-

Factorial of 10 = 3628800

Q.5.

Two numbers are entered through the keyboard. Write a program to find the value of one number raised to the power of another. (Do not use Java built-in method)

```
import java.util.Scanner;
public class Power_of_Number {
    public static void main (String args[]) {
        Scanner sc = new Scanner (System.in);
        System.out.println ("Enter the base
                            number:");
        int base = sc.nextInt();
        int temp = base;
        System.out.println ("Enter the exponent
                            number:");
        int exp = sc.nextInt();
        for (int i = 1; i <= exp; i++) {
            temp = temp * temp;
        }
        System.out.println ("Result of " + base +
                            " power " + exp + " is " + temp);
```

3
}

Q.6. write a program that prompts the user to input an integer and then outputs the number with the digits reversed. for example, if the input is 12345, the output should be 54321

```
import java.util.Scanner;
```

```
class Reverse_Number
```

```
{
```

```
public static void main (String args[])
```

```
{
```

```
int num = 0;
```

```
int reversenum = 0;
```

```
System.out.println ("Enter the number");
```

```
Scanner in = new Scanner (System.in);
```

```
num = in.nextInt();
```

```
while (num != 0)
```

```
{
```

```
reversenum = reversenum * 10;
```

```
reversenum = reversenum + num % 10;
```

```
num = num / 10;
```

```
}
```

```
System.out.println ("Reverse number is:");
```

```
}
```

```
}
```

Output:

Enter the number: 12345

reverse number is: 54321

Q.7 Write a program that reads a set of integers, and then prints the sum of the even and odd integers.

```
import java.util.Scanner;
public class Sum-Odd-Even
{
    public static void main(String args[])
    {
        int n, sumE = 0, sumO = 0;
        Scanner s = new Scanner(System.in);
        System.out.println("Enter the number");
        n = s.nextInt();
        int[] a = new int[n];
        System.out.println("enter the numbers that you want to enter");
        for (i=0; i<n; i++)
        {
            a[i] = s.nextInt();
        }
        for (i=0; i<n; i++)
        {
            if (a[i] % 2 == 0)
            {
                sumE = sumE + a[i];
            }
            else
                sumO = sumO + a[i];
        }
        System.out.println("The sum of even numbers is " + sumE);
        System.out.println("The sum of odd numbers is " + sumO);
    }
}
```

```
sumo = sumo + a[i]; }  
}
```

```
System.out.println("even sum = " + sume);
```

```
System.out.println("odd sum = " + sumo);
```

```
}
```

```
}
```

Q.8. write a program that prompts the user to input a positive integer. It should then output a message indicating whether the number is a prime number.

```
class Primenro  
{
```

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

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

```
int flag = 0;
```

```
for (int i = 2; i < num; i++)  
{
```

```
if (num % i == 0)  
{
```

```
System.out.println (num + " is not a  
prime Number");
```

```

    Flag = 1;
    break;
}
}
else if (flag == 0)
System.out.println(num + " is a prime
Number");
}
}

```

Output:

5

5 is a prime Number.

Q.9. write a program to calculate HCF of
Two given numbers

```

import java.util.Scanner;
public class HCFOfTwoNumbers {
    public static void main(String args[])
    {
        int a, b, i, hcf = 0;
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter first number:");
        a = sc.nextInt();

```

```
System.out.println("Enter second number");
b = sc.nextInt();
for (i=1; i<=a || i<=b; i++) {
    if (a % i == 0 && b % i == 0)
        hcf = i;
}
System.out.println("HCF of given two numbers is: " + hcf);
```

Output:

Enter first number:

625

Enter second number:

125

HCF of given two numbers is: 125

Q.10. Write a do-while loop that asks the user to enter two numbers. The numbers should be added and the sum displayed. The loop should ask the user whether he or she wishes to perform the operation again. If so, the loop should repeat, otherwise it should terminate.

```
import java.util.*;
public class sumDemo
{
    public static void main(String args[])
    {
        int number1, number2;
        char choice;
        Scanner scn = new Scanner(System.in);
        do
        {
            System.out.print("Enter first number:");
            number1 = scn.nextInt();
            System.out.print("Enter second number:");
            number2 = scn.nextInt();
            System.out.print("sum of the both numbers is: " + (number1 + number2));
            System.out.print("\nDo you want to perform the operation again? press Y/y:");
            choice = scn.next().charAt(0);
        } while (choice == 'y' || choice == 'Y');
```

Q.11. write a program to enter the numbers till the user wants and at the end the program it should display the count of positive, negative and zeros entered.

```
import java.util.Scanner;
```

```
public class Javaprogram
```

```
{
```

```
    public static void main(String args)
```

```
    {
```

```
        int countp=0, countn=0, countz=0;
```

```
        int arr[] = new int[10];
```

```
        Scanner scan = new Scanner(System.in);
```

```
        System.out.println("Enter 10 Numbers");
```

```
        for (i=0; i<10; i++)
```

```
{
```

```
            arr[i] = scan.nextInt();
```

```
}
```

```
        for (i=0; i<10; i++)
```

```
{
```

```
            if (arr[i] < 0)
```

```
{
```

```
                countn++;
```

```
}
```

```
else if (arr[i] == 0)
```

```
{
```

```
    countz++;
```

```
}
```

```
else
```

```
{
```

```
    countp++;
```

```
}
```

```
}
```

```
System.out.println(countp + " positive  
numbers");
```

```
System.out.println("n" + countn + " negative  
numbers");
```

```
System.out.println("n" + countz "zero");
```

```
{
```

```
}
```

output : Enter 10 Number : 12

-6

67

0

9

0

-6

-4

6

28

5 positive numbers

3 Negative Numbers

2 zero

Q.12. write a program to print enter the numbers till the user wants and at the end the program should display the largest and smallest numbers entered

```
import java.util.Scanner;
public class numbers
{
    public static void main (String args[])
    {
        Scanner console = new Scanner(System.in);
        int number;
        int max = Integer.MIN_VALUE;
        int min = Integer.MAX_VALUE;
        char choice;
        char do
        {
            System.out.println ("Enter the number");
            number = console.nextInt();
            if (number > max)
            {
                max = number;
            }
            if (number < min)
            {
                min = number;
            }
        }
    }
}
```

{ while (ch

System.out.println("Do you want to continue y/n ?");

choice = console.next().charAt(0);

{ while (choice == 'y' || choice == 'Y');

System.out.println("Largest number :" +
max);

System.out.println("smallest number :" +
min);

}

output:

Enter the number: 5

Do you want to continue y/n? . y

Enter the number: 2

Do you want to continue y/n? y

Enter the number: 10

Do you want to continue y/n? n

Largest number : 10,

smallest number : 2

Q.13. write a program to print out all

Armstrong numbers between 1 and 500

if sum of cubes of each digit of the
number is equal to the number itself
then the number is called, an Armstrong
number.

for example, $153 = (1 * 1 * 1) + (5 * 5 * 5)$
 $= (3 * 3 * 3)$

```
import java.util.Scanner;
public class ArmstrongNumbers
{
    public static void main (String args[])
    {
        int num1, num2;
        Scanner sc = new Scanner (System.in);
        System.out.println ("Enter the first no.");
        num1 = sc.nextInt();
        System.out.println ("Enter the second no.");
        num2 = sc.nextInt();
        for (int i = num1; i <= num2; i++)
        {
            int check, rem, sum = 0;
            check = i;
            while (check != 0)
            {
                rem = check % 10;
                sum = sum + (rem * rem * rem);
                check = check / 10;
            }
            if (sum == i)
            {
                System.out.println ("+" + i + " is an
Armstrong number.");
            }
        }
    }
}
```

Output:

Enter the first number: 1

Enter the second no : 500

1 is an Armstrong number.

153 is an Armstrong number

370 is an Armstrong number

371 is an Armstrong number

407 is an Armstrong number

Q.14. write a program to print Fibonacci series of n terms where n is input by user: 0 1 1 2 3 5 8 13 24

```
class Fibonacci {
```

```
public static void main (String args[])
```

```
{
```



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

```
System.out.println ("Fibonacci series");
```

```
int f1, f2 = 0, f3 = 1;
```

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

```
{
```

```
System.out.println (" " + f3 + " ");
```

```
f1 = f2;
```

```
f2 = f3;
```

```
f3 = f1 + f2;
```

```
3
```

```
77
```

Output : 6

Fibonacci Series

1 1 2 3 5 8

Q.15. writing a program to calculate the sum of following series where n is input by user.

$$1 + 1/2 + 1/3 + 1/4 + 1/5 + \dots + 1/n$$

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

```
class GF6
```

```
{
```

```
static double sum(int n)
```

```
{
```

```
    double i, s = 0.0;
```

```
    for (i = 1; i <= n; i++)
```

```
        s = s + 1/i;
```

```
    return s;
```

```
}
```

```
public static void main(String  
args[])
```

```
{
```

int n = 5;

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

}

}

100% makes below situation

100% 100%

int n) example (0,3) sum formula

(= 0 + 1 + 2 + 3)

so 100% can be shown

100% notes() storing 100% above

100% 100%

100% addition

(add 100% addition)

100% 100% addition

100% 100% addition

Q.16.

compute the natural logarithm of 2 by adding up to n terms in the series $1 - \frac{1}{2} + \frac{1}{3} - \frac{1}{4} + \frac{1}{5} \dots \frac{1}{n}$.

```
import java.util.*;  
public class log  
{  
    public static void main (String  
        args [])  
    {
```

```
Scanner sc = new scanner (System.in)  
int l, n, sign = -1
```

```
double sum = 0;
```

```
System.out.println ("enter the value of  
n");
```

```
n = sc.nextInt ();  
for (i=1; i<=n; i++)
```

```
{
```

```
sign * = -1;
```

```
sum + = sign * 1.0 / i;
```

System.out.println("log₂: " + tsum)

for (int i = 0; i < n; i++)

System.out.println("2^{" + i + "} = " + 2 * 2 * 2 * 2 * 2 * 2 * 2 * 2)

2⁰ = 1
2¹ = 2
2² = 4
2³ = 8
2⁴ = 16
2⁵ = 32
2⁶ = 64
2⁷ = 128
2⁸ = 256
2⁹ = 512
2¹⁰ = 1024
2¹¹ = 2048
2¹² = 4096
2¹³ = 8192
2¹⁴ = 16384
2¹⁵ = 32768
2¹⁶ = 65536
2¹⁷ = 131072
2¹⁸ = 262144
2¹⁹ = 524288
2²⁰ = 1048576
2²¹ = 2097152
2²² = 4194304
2²³ = 8388608
2²⁴ = 16777216
2²⁵ = 33554432
2²⁶ = 67108864
2²⁷ = 134217728
2²⁸ = 268435456
2²⁹ = 536870912
2³⁰ = 1073741824
2³¹ = 2147483648
2³² = 4294967296
2³³ = 8589934592
2³⁴ = 17179869184
2³⁵ = 34359738368
2³⁶ = 68719476736
2³⁷ = 137438953472
2³⁸ = 274877906944
2³⁹ = 549755813888
2⁴⁰ = 1099511627776
2⁴¹ = 2199023255520
2⁴² = 4398046511040
2⁴³ = 8796093022080
2⁴⁴ = 17592186044160
2⁴⁵ = 35184372088320
2⁴⁶ = 70368744176640
2⁴⁷ = 140737488353280
2⁴⁸ = 281474976706560
2⁴⁹ = 562949953413120
2⁵⁰ = 1125899906826240
2⁵¹ = 2251799813652480
2⁵² = 4503599627304960
2⁵³ = 9007199254609920
2⁵⁴ = 18014398509219840
2⁵⁵ = 36028797018439680
2⁵⁶ = 72057594036879360
2⁵⁷ = 144115188073758720
2⁵⁸ = 288230376147517440
2⁵⁹ = 576460752295034880
2⁶⁰ = 1152921504590069760
2⁶¹ = 2305843009180139520
2⁶² = 4611686018360279040
2⁶³ = 9223372036720558080
2⁶⁴ = 18446744073441116160
2⁶⁵ = 36893488146882232320
2⁶⁶ = 73786976293764464640
2⁶⁷ = 147573952587528929280
2⁶⁸ = 295147905175057858560
2⁶⁹ = 590295810350115717120
2⁷⁰ = 1180591620700231434240
2⁷¹ = 2361183241400462868480
2⁷² = 4722366482800925736960
2⁷³ = 9444732965601851473920
2⁷⁴ = 18889465931203702947840
2⁷⁵ = 37778931862407405895680
2⁷⁶ = 75557863724814811791360
2⁷⁷ = 151115727449629623582720
2⁷⁸ = 302231454899259247165440
2⁷⁹ = 604462909798518494329840
2⁸⁰ = 1208925819597036988659680
2⁸¹ = 2417851639194073977319360
2⁸² = 4835703278388147954638720
2⁸³ = 9671406556776295909277440
2⁸⁴ = 19342813113552591818554880
2⁸⁵ = 38685626227105183637109760
2⁸⁶ = 77371252454210367274219520
2⁸⁷ = 154742504908420734548439040
2⁸⁸ = 309485009816841469096878080
2⁸⁹ = 618970019633682938193756160
2⁹⁰ = 1237940039267365876387512320
2⁹¹ = 2475880078534731752775024640
2⁹² = 4951760157069463505550049280
2⁹³ = 9903520314138927011100098560
2⁹⁴ = 19807040628277854022200197120
2⁹⁵ = 39614081256555708044400394240
2⁹⁶ = 79228162513111416088800788480
2⁹⁷ = 158456325226222832177601576960
2⁹⁸ = 316912650452445664355203153920
2⁹⁹ = 633825300904891328704006307840
2¹⁰⁰ = 1267650601809782657408012615680

Q.17. write a program that generates a random number and asks the user to guess what the number is. if the user's guess is higher than the random number , the program should display "Too high , try again." If the user's guess is lower than the random number , the program should display "Too low , try again." The program should use a loop that repeats until the user correctly

```
import java.util.Scanner;
```

```
import java.util.Random;
```

```
public class GuessingGame
```

```
{
```

```
    public static void main(String args  
                           [] )
```

```
{
```

```
    Scanner kb = new Scanner (System.in);  
    Random rand = new Random();
```

```
    int num = rand.nextInt(100)+1;
```

```
    int guess = 0;
```

```
    int count = 0;
```

```
int guesses = 0;
```

```
do
```

```
{
```

```
System.out.println("Guess what number  
I have (1-100)?");
```

```
guess = kb.nextInt();
```

```
if (num > guess)  
{
```

```
System.out.println("Too high, try again");
```

```
}
```

```
else if (num < guess)  
{
```

```
System.out.println("Too low, try again");
```

```
}
```

```
System.out.println("you're right, the  
number is " + num);
```

```
System.out.println("you guessed " + gue-  
sses + " times");
```

```
}
```

```
}
```

```
while (guess != num);
```

```
}
```

```
}
```

Q.18. write a program to print following

i) * * * * *
* * * * *
* * * * *
* * * * *

class output

```
public static void main (String args[])
```

```
{
```

```
    int i, j;  
    for (i = 1; i <= 5; i++)
```

```
        for (j = 1; j <= i; j++)
```

```
            System.out.println ("*");
```

```
    }
```

```
}
```

```
}
```

iii)

```
*  
**  
***  
****  
***** class Star {  
public static void printStars (int n)  
{  
    int i, j;  
    for (i=0; i<n; i++)  
    {  
        for (j=0; j<=i; j++)  
        {  
            System.out.print ("*");  
        }  
        System.out.println ();  
    }  
}
```

iii) *

* *

* * *

* * * *

* * * *

```
class Star {  
    public static void printStars(int n)  
    {  
        int i, j;  
        for (i = 0; i < n; i++)  
        {  
            for (j = 2 * (n - i); j >= 0; j--)  
            {  
                System.out.print(" ");  
            }  
            for (j = 0; j <= i; j++)  
            {  
                System.out.print("*");  
            }  
            System.out.println();  
        }  
    }  
}
```

iv) *

* * *

* * * * *

* * * * *

* * * * *

```
public class Recstar
```

```
{
```

```
public static void main(String args)
```

```
{
```

```
int i, j;
```

```
for (i=1; i<=4; i++)
```

```
{
```

```
for (j=1; j<=10; j++)
```

```
{
```

```
System.out.println('*');
```

```
}
```

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

```
}
```

```
}
```

```
}
```

v)

1

2 2 2

3 3 3 3

4 4 4 4 4 4

5 5 5 5 5 5 5 5

```
import java.util.Scanner;
```

```
public class Char demo
```

```
{
```

```
public static void main (String args[])
{
    Scanner sc = new Scanner (System.in);
    System.out.println ("Enter the number of rows:");
    int rows = sc.nextInt();
    for (int i=1; i<=rows; i++)
    {
        for (int j=1; j<=i; j++)
        {
            System.out.print (i);
        }
        System.out.println ();
    }
    sc.close();
}
```

vi)

1

2 1 2

3 2 1 2 3

4 3 2 1 2 3 4

5 4 3 2 1 2 3 4 5

```
public class one
```

```
{
```

```
public static void main (String args  
[ ])
```

```
{
```

```
int i, j, k;
```

```
for (i = 1; i <= 5; i++)
```

```
{
```

```
for (j = 5; j > i; j--)
```

```
{
```

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

```
}
```

```
for (k = 1; j <= 2 * i - 1; k++)
```

```
{
```

```
System.out.print ( "+ " ) ;
```

```
}
```

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

```
}
```

```
{
```

```
{
```

Q.19. Write a program to compute ~~sign~~
for given x . The user should supply
 x and a positive integer n . We com-
pute the cosine of x using the series
and the computation should use all
terms in the series up through the
term involving x^n .

$$\sin x = x - x^3/3! + x^5/5! - x^7/7! + \dots$$
$$\cos x = 1 - x^2/2! + x^4/4! - x^6/6! + \dots$$

```
import java.util.*;  
public class Cos  
{
```

```
    public static void main (String  
        args[])
```

```
    {  
        Scanner sc = new Scanner (System.in);  
        int i, j, n, fact, sign = -1;  
        float x, p, sum = 0;
```

```
        System.out.println ("enter the value of x");  
        x = sc.nextInt();
```

```
        System.out.println ("enter the value of n");  
        n = sc.nextInt();
```

```
        for (i=1; i<=n; i+=2)  
        {
```

```
            p = 1;
```

```
            fact = 1;
```

for(j=1; j<=i; j++)

p = p * x[j]

fact = fact * j;

}

sum += sign * p / fact;

sign = -1 * sign;

}

System.out.println("sinx = "+(1+sum));

}

}

else

Q.20. write a program to compute the value of $\cos x$. The user should supply x and a positive integer n . we compute the cosine of x using the series and the computation should use all terms in the series up through the term involving x^n

$$\cos x = 1 - x^2/2! + x^4/4! - x^6/6! \dots$$

$$\sin x = x - x^3/3! + x^5/5! - x^7/7! + x^9/9! \dots$$

```

import java.util.*;
public class sin
{
    public static void main (String args[])
    {
        Scanner sc = new Scanner (System.in);
        int i, j, n, fact, sign = -1;
        float x, p, sum = 0;
        System.out.println ("enter the value of x = ");
        x = sc.nextInt();
        System.out.println ("enter the value of n = ");
        n = sc.nextInt();
        for (i = 1; i <= n; i += 2)
        {
            p = 1;
            for (j = 1; j <= i; j++)
                p *= j;
            fact = p;
            if (i % 4 == 0)
                sign = -1;
            else if (i % 4 == 1)
                sign = 1;
            sum += sign * (x * x) / fact;
        }
        System.out.println ("the value of sin x is " + sum);
    }
}

```

$p = 1;$

$fact = 1;$

for ($j = 1; j \leq i; j++$)

{

$p = p * x;$

$fact = fact * j;$

}

$sign = -1 * sign;$

$sum += sign * p / fact;$

}

System.out.println("sin = " + sum);

}