ASSIGNMENT 1.

1. Check whether the given number is armstrong or not

A:

**package** org.jsp.app;

**import** java.util.Scanner;

**public** **class** Armstrong

{

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

{

Scanner sc=**new** Scanner(System.***in***);

System.***out***.println("Enter a number : ");

**int** num=sc.nextInt();

**int** rem,temp,sum = 0;

temp = num;

**while** (num > 0)

{

rem= num % 10;

sum = sum + rem\*rem\*rem;

num /= 10;

}

**if**(temp == sum)

System.***out***.println(temp + " is an Armstrong number");

**else**

System.***out***.println(temp + " is not an Armstrong number");

}

}

OUTPUT : Enter a number :

153

153 is an Armstrong number

1. Find out all the Armstrong numbers falling in the range of 100-999.

**import** java.util.Scanner;

**public** **class** Armstrongnum

{

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

{

**for**(**int** i=100;i<=1000;i++)

{

**int** num, rem,sum = 0;

num=i;

**while** (num > 0)

{

rem= num % 10;

sum = sum + rem\*rem\*rem;

num /= 10;

}

**if**(i == sum)

{

System.***out***.println(i + " is an Armstrong number");

}

}

}

}

OUTPUT : 153 is an Armstrong number

370 is an Armstrong number

371 is an Armstrong number

407 is an Armstrong number

1. Find Simple and compound interest ?

A:

public class Interest {

public void simpleInterest(double rate, int time, int principal){

double SI = (principal\* rate \*time) /100;

System.out.println("Simple Interest is: " +SI);

}

public void compoundInterest(int principal , int time , double rate, int n ){

double amount = principal \*Math.pow(1+(rate/n),n \*time);

double CI= amount-principal;

System.out.println("compound Interest is: " +CI);

}

public static void main(String arg[]){

Interest i= new Interest();

i.simpleInterest(1000, 5, 2);

i.compoundInterest(3000,5,8,12);

}}

1. Supply mark of three subject and declare the result based on conditions.

A:

Public class calculateMarks{

Public static void main(String[] arg){

Scanner sc = new Scanner(System.in);

Scanner sc=**new** Scanner(System.***in***);

System.***out***.println("Enter the marks in subject1 :");

**double** sub1=sc.nextInt ();

System.***out***.println("Enter the marks in subject2 :");

**double** sub2=sc.nextInt ();

System.***out***.println("Enter the marks in subject3 :");

**double** sub3=sc.nextInt ();

**if**(sub1>60 && sub2>60 &&sub3>60)

{

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

}

**else** **if**((sub1>60 && sub2>60)||(sub2>60 && sub3>60)||(sub1>60 &&sub3>60))

{

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

}

**else**

{

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

}

}

}

OUTPUT: Enter the marks in subject1 :

85

Enter the marks in subject2 :

90

Enter the marks in subject3 :

70

Pass

5.Calculate the income tax on the basis of the following table.

Note:-Assume slab is consider for Male, Female as well as Senior citizen

| **Slab** | **Income range** | **Tax Payable in percent** |
| --- | --- | --- |
| Slab A | 0 - 1,80,000 | Nil |
| Slab B | 1,80,001 - 3,00,000 | 10% |
| Slab C | 3,00,001 - 5,00,000 | 20% |
| Slab D | 5,00,001 - 10,00,000 | 30% |

1. Accept CTC from user and display tax amount

**public** **class** IncomeTax

{

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

{

**double** tax = 0, CTC;

Scanner scanner = **new** Scanner(System.***in***);

System.***out***.println("Enter income : ");

CTC = scanner.nextDouble();

**if**(CTC <= 180000)

{

tax = 0;

}

**else** **if**(CTC > 180000 && CTC <= 300000)

{

tax = (CTC/100)\*10;

System.***out***.println("Income tax payable is : " + tax);

}

**else** **if**(CTC > 300000 && CTC <= 500000)

{

tax = (CTC/100)\*20;

System.***out***.println("Income tax payable is : " + tax);

}

**else** **if**(CTC > 500000 && CTC <= 1000000)

{

tax = (CTC/100)\*30;

System.***out***.println("Income tax payable is : " + tax);

}

}

}

OUTPUT :

Enter income :

1000000

Income tax payable is : 300000.0

1. Consider a CUI based application, where you are asking a user to enter his Login name and password, after entering the valid user-id and password it will print the message “Welcome” along with user name. As per the validation is concerned. The program should keep a track of login attempts. After three attempts a message should be flashed saying “Contact Admin” and the program should terminate.

A:

7. There is an Array which is of the size 15, which may or may not be sorted. you should write a program to accept a number and search if it in contained in the array

* Example:

| **Array Elements** | **5** | **12** | **14** | **6** | **78** | **19** | **1** | **23** | **26** | **35** | **37** | **7** | **52** | **86** | **47** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Indexes | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |

* Value to be search is 19

**import** java.util.Arrays;

**import** java.util.Scanner;

**public** **class** Array

{

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

{

**int** arr[] = {5,12,14,6,78,19,1,23,26,35,37,7,52,86,47};

System.***out***.println(Arrays.*toString*(arr));

Scanner scanner = **new** Scanner(System.***in***);

System.***out***.println("Enter a number to search in array : ");

**int** n = scanner.nextInt();

**for**(**int** i = 0; i< arr.length; i++)

{

**if**(arr[i] == n)

{

System.***out***.println(n + " is found in the array at " + i + "th index.");

}

}

}

}

OUTPUT :

[5, 12, 14, 6, 78, 19, 1, 23, 26, 35, 37, 7, 52, 86, 47]

Enter a number to search in array :

19

19is found in the array at 5th index

8.Bubble sort

**import** java.util.Arrays;

**public** **class** BubbleSort

{

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

{

**int** arr[] = {5,12,14,6,78,19,1,23,26,35,37,7,52,86,47}, temp;

System.***out***.println(Arrays.*toString*(arr));

**for**(**int** i = 0; i < arr.length-1; i++)

{

**for**(**int** j = 0; j < arr.length - i - 1; j++)

{

**if**(arr[j] > arr[j+1])

{

temp = arr[j];

arr[j] = arr[j+1];

arr[j+1] = temp;

}

}

}

System.***out***.println("Sorted Array :");

System.***out***.println(Arrays.*toString*(arr));

}

}

OUTPUT :

[5, 12, 14, 6, 78, 19, 1, 23, 26, 35, 37, 7, 52, 86, 47]

Sorted Array :

[1, 5, 6, 7, 12, 14, 19, 23, 26, 35, 37, 47, 52, 78, 86]