10-7-2024

K.Meghana
192321126
CSA0985
JAVA

Ⓟ

10/07/2024

Assignment-01

1. Sum of natural numbers.

```
Public class sum of naturalNumber_
{
Public static void main (string [] args)
{
int i; num=10; sum=0;
for (i=1 ; i<=num ; ++i)
{
sum =sum+i;
}
system out.Printn ("sum of first 10NaturalNumbers is ="+ sum);
}
}
```

Output:

```
Sum of first 10NaturalNumber is = 55
```

2. Given number it is Palindrome number or not

```
class Palindrome Example{
  Public static voidmain(string args[]){
    int r; sum=0; temp;
      int n=454;
      temp=n;
    while (n>0){
      r=n%10
    sum =(sum*10)+r;
      n=n/10;
  }
  if(temp ==sum)
system.Out Printn ("palindrome number");
else
8 system out Printin("not Palindrome");
   }
 }
```

Output

Palindrome Number

3. Given number it is Prime Number or not.

```
   Public class main{
   Public static void main (string[]args){
       int num =29;
     boolean flag = false;
     for(int i=2 ; i<=num/2 ; ++i){
       if (num%i ==0){
          flag =true;
       break;
      }
     }
   if (!flag)
   system.out.Printin (num + "is a Prime number.")
 else
   system.out.print in (num+ "is not a Prime number.")
  }
 }
```

Output

29 is a Prime number

4. N factorial of a Number

```
  class factorial Example{
Public static void main(string arg[]){
    int i, fact=1;
      int number=5;
   for( i=1; i<=number; i++){
        fact =fact*i;
  }
  system.out.Println("factorial of "+number+"is:"+fact);
  }
}
```

Output

```
factorial of 5 is: 120
```

5. Reverse a number

```
  Public class reverse number example
{
 Public static void main(string[] args)
{
int number=987654, reverse=0;
 while (number!=0)
{
 int remainder=number%10;
   reverse =reverse*10 + remainder.
     number=number/10;
 }
system out println ("The reverse of the given number is;"reverse)
 }
}
```

Output

```
456789
```

6. Armstrong numbers:

```
  static boolen is Armstrong (int n)
    {
  int temp digits =0, Last =0, sum =0;
    temp=n;
  while (temp>0)
  {
  temp = temp/10;
  digits++;
  }
  temp =n;
  while (temp>0)
 {
 Last = temp%10
 sum+= (Math. Pow(Last.digits));
  temp =temp/10;
 }
 if (n==sum)
  return true;
else return false;
}
Public static void main (string args[])
{
~~int main~~;
 int num;
scanner sc =news canner(system.in);
system.out.print ("Enter the Limit");
  num=sc.nextint(8);
system out print ("Armstrong Number upto "+num+"are)
  for(int i=0; i<num; i++)
     if (is Armstrong(i))
 system out print (i+",");
 }
}
```

Output Enter=999

Armstrong number upto 999 are

0,1,2,3,4,5,6,7,8,9,153,370,371,407

Happy number

```
def is HappyNumber(num);
    rem =sum=0;
  while (num>0);
   rem =num%10;
    sum=sum + (rem*rem);
    num =num//10;
    return sum;
    num=82;
    result=sum;
 while (result != 1 and result != 4):
   result =is Happy Number (result);
    if (result==1);
      Print(str(num)+"is a happy number");
 elif(result==4);
 Print(str(num)+"isnot a happy number");
```

Output

```
82 is a happy number
```

8. Sum of the digits

```
Public static void main (string args[])
{
int number, digitsum=0;
scanner sc=newscanner(System;in);
 system.out.print("Enter the number:");
  number =sc.next.int();
  while (number>0)
{
digit= number%10;
sum=sum+digit;
number =number%10;
}
system.out.Println ("sum of digits;"+sum);
}
Output
Enter the number :876
   sum of digits ;21
```

9. Number divisible by 5 and 7

```
  import java util.
   class GFG {
  static int Number (int n)
 {
 for (int i=1 ; i<n+1 ; i++)
 {
 if (i%5 ==0 ||i% 7 ==0)
    system out print (i+" ");
 }
 return;
}
Public static void main (string args[])
{
int N =50;
 NUM Gem (N);
}

}
```

Output:

```
5 7 10  14 15 20 21 25 28 30 35 40 42 45 49 50
```

Perfect Number

```
 Public static void main (string args[])
{
 long n. sum =0;
 scanner sc = new scanner (system.in);
 system.out.Print("enter the number");
   n=sc nextlong();
   int i=1;
 while (i<=n/2)
 {
 if(n%i==0)
 {
 sum=sum+i;
 }
  i++;
 }
 if(sum==n)
 {
system.out.print(n+"is a Perfect number.");
}
else
system.out.println(n+"is not a Perfect number.");
}
```

Output.

```
enter :28
28 is a Perfect number.
```

GCD and LCM

```
 import java.io.*
 class GFG {
 startic int gcd (int a, intb) {
    if(b==0)
   return a;
 else
    return gcd(b,a%b);
 }
 static int LCM(int a, int b, int gcdvalue)
 {
  return math.abs(a*b)/gcdvalue;
 }
 Public static void main(string[] args){
    int a=20, b=30, gcd value;
    gcd value = gcd(a,b);
  system out print in("GCD = "+gcd value);
 system out println("LCM = "+ lcm(a,b,gcdvalue));
  }
```

Output

```
GCD=10
LCM=60
```

12 Decimal to binary

```
    Static void dectobinary(int n)
 {
 int[] binaryNum = new int[1000];
    int i=0;
  while(n>0)
 {
 binaryNum[i] =n%2;
   n=n/2;
  i++;
 }
for(int j=i-1; j>=0; j--)
system out.print(binaryNum[j]);
}
Public static void main(string[] args)
{
 int n=17;
 system.out.Println("Decimal-"+n);
 system.out.Print("Binary-");
 dec to Binary(n);
 }
}
```

Output

```
Decimal - 17
Binary - 10001
```

13. Binary to decimal

```
 class GFG{
 static int binary to decimal(int n)
 {
  int num=n;
  int dec-value=0;
  int base=1;
  int temp=num;
  while(temp>0){
  int last_digit=temp%10
  temp=temp/10;
 dec-value+=last_digit*base;
  base=base*2;
 }
return dec_value;
}
Public static void main(string[]args)
 {
 int num=10101001;
system.out.printin (binary to decimal(10101001));
 }
```

Output

169

14 Celsius to Fahrenheit

```
  Public class temperature
 {
 Public static void main (string args[])
 {
  float Fahrenheit, celsius;
    celsius=13;
  Fahrenheit=((celsius*9)/5)+32;
 system.out.println("Temperature in fahrenheit is="+ fahrenheit);
 }}
```

Output

Temperature in fahrenheit is: 55.4

15 Sum of odd numbers and even numbers

```
  import java.io.*;
  Public class GFG{
   Public static void main (string [] args)
 {
 int n=8;
 int even sum=0;
int add sum=0;
for(int i=1; i<=2*n; i++){
  if((i&1)==0)
  evensum +=i;
else
 Oddsum +=i;
}
system.out.Println("sum of first"+n+"even numbers ="+evensum);
system.out.Println("sum of first"+n+"odd numbers ="+odd sum);
 }
}
```

Output:

```
sumof first 8 even numbers =72
sum of first 8 odd numbers =64
```

16. Even orodd

```
 class even_odd{
   Public static voidmain(string[] args)
 {
 int n=5; even sum =0, oddsum =0;
 for(int i=1; i<=2*n; i++){
 if((i&1)==0)
 even sum +=i;
 else
 odd sum +=i;
}
system.out.Println("sum of first"+n+"even numbers="+evensum);
system.out.println("sumof first"+n+"odd numbers ="+oddsum);
 }
}
```

Output

```
Sum of first 8 even numbers =72
Sum of first 8 odd numbers =84
```

17 Voting

```
class voting age
{
Public static void main(string[] args)
{
int age;
scanner sc = new scanner(system.in);
  system.out.print("Enter your age");
    age = sc.nextInt();
    if (age >= 18)
{
system.out.Println ("you are eligible for vote.");
}
else
{
system.out.Println ("you are not eligible for vote.");
  }
 }
}
```

Output

```
Enter = 19
you are eligible for vote
```

18 Vowels and consonants

```
  class vowel consonants{
  Public static void main(string[] args){
    char ch = 'i';
 if (ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u')
 system.out.Println(ch + "is vowel");
else
 system.out.Println(ch + "is consonant");
```

```
  }
 }
```

output

1 is vowel

19 strong number

```
   class test {
  Public static void main (String[]args)
  {
  int sum =0, i, r, fact, number;
   scanner(system.in);
  system out.println ("enter a number:");
    number = in.nextInt();
    int Original = number;
  while (number>0)
 {
  r=number % 10;
   fact=1;
  for(i=1; i<r; i++)
  {
 fact= fact*i;
}
sum =sum + fact;
number =number/10;
}
if (original ==sum)
system.out.Println("strong Number");
else
system.out.println("Not strong number");
 }
}
```

output

145

20 squareroot & cube root

```
import java.util.scanner;
  Public class RootsCalculator {
       Public static void main(string[] args) {
 Scanner scanner = new scanner(system.in);
   System.out.Println("Enter a number:");
  double number = scanner.next Double();
   double squareRoot = Math.sqrt(number);
   double cubicRoot = Math.cbrt(number);
   System.out.Println(("square root of" + number + " is:" + squareroot);
System.out.Println("Cubic root of" + numbr + " is:" + cubicRoot);
 }

}
```

Output:

$4^2 = 16$

$2^3 = 8$