DIGIT HANDLING PROJECT

Created by Sanjay Sir of SOFTECH on 30/05/2010

Home
Project home

```
import java.io.*;
public class digit
{
int choice;
BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
public void inputmenu()throws IOException
{
 System.out.println("1. Automorphic number");
 System.out.println("2. Magic Number");
 System.out.println("3. Krishnamurthy Number");
 System.out.println("4. Armstrong Number");


 System.out.println("Enter your choice (1-4)");
 choice=Integer.parseInt(br.readLine());
 if(choice<0||choice>4)
 {
  System.out.println("You have entered a wrong choice...program terminated");
  System.exit(0);
 }

 System.out.println("Enter the number");
 int num=Integer.parseInt(br.readLine());
 digitchecking(num);  // calling digitchecking function
}
public void digitchecking(int n)throws IOException
{
 switch(choice)
 {
  case 1:
  boolean ch=isAutomorphic(n);
  if(ch==true)System.out.println(n+" is an Automorphic number");
  else System.out.println(n+" is not an Automorphic number");
  break;

  case 2:
  boolean m=isMagic(n);
  if(m==true)System.out.println(n+" is a magic number");
  else System.out.println(n+" is not a magic number");
  break;

  case 3:
  boolean k=isKrishnamurthy(n);
  if(k==true)System.out.println(n+" is a Krishnamurthy number");
  else System.out.println(n+" is not a Krishnamurthy number");
  break;

  case 4:
  boolean A=isArmstrong(n);
  if(A==true)System.out.println(n+" is an Armstrong number");
  else System.out.println(n+" is not an Armstrong number");
  break;

  //default: System.out.println("Wrong choice"); //this is already checked before hence not required
 }//end of switch case
}//end of digitchecking function

public boolean isAutomorphic(int n)
{
 int sq,copy,rvalue,count=0;
 boolean found=false;
 sq=n*n;  //storing square of the number
 copy=n;
 while(copy>0)
 {
  copy/=10;
  count++;
 }
 rvalue=sq%(int)Math.pow(10,count);  //extracting last part of the number
 if(rvalue==n)
 {
  found=true;
 }

 return found;
}//end of isAutomorphic function

public boolean isMagic(int n)
{
  int d,s;
  boolean check=false;
  do
  {
    s=0;
    while(n>0)
```

```
    {
     d=n%10;
     s=s+d;
     n=n/10;
    }
    n=s;
   }while(s>9);

  if(s==1)
  check=true;

  return check;
}//end of isMagic function

public boolean isKrishnamurthy(int n)
{
  int d,s=0,f,copy=n;

  boolean check=false;


  while(copy>0)
  {
   d=copy%10;

   f=1;
   for(int i=1; i<=d; i++)
   {
    f=f*i;  //calculating factorial
   }
   s=s+f;
   copy=copy/10;
  }
  if(s==n)
  check=true;

  return check;
}//end of isKrishnamurthy function

public boolean isArmstrong(int n)
{
  int d,s=0,copy=n;
  boolean chk=false;
  while(copy>0)
  {
   d=copy%10;
   s=s+d*d*d;
   copy/=10;
  }
  if(s==n)
  chk=true;

  return chk;
}//end of isArmstrong function

public void main()throws IOException
{
 inputmenu();
}//end of main function
}//end of class digit
```