2.Armstrong within a limit

```
import java.util.Scanner;
public class Armstrong {
  public static void main(String args[]){
    int num;
    Scanner sc = new Scanner(System.in);
    System.out.println("Enter the limit:");
    num = sc.nextInt();

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

3.Base Conversion

```
import java.util.Scanner;
class Dec_conv
{
  public static void main(String args[])
   {
      Convert obj = new Convert();
       obj.getVal();
       obj.convert();
   }
}
class Convert
{
   Scanner scan;
   int num;
   void getVal()
    {
        System.out.println("Decimal to HexaDecimal,Octal and Binary");
        scan = new Scanner(System.in);
        System.out.println("\nEnter the number :");
        num = Integer.parseInt(scan.nextLine());
    }
```

```
    void convert()
     {
         String hexa = Integer.toHexString(num);
         System.out.println("HexaDecimal Value is : " + hexa);
         String octal = Integer.toOctalString(num);
         System.out.println("Octal Value is : " + octal);
         String binary = Integer.toBinaryString(num);
         System.out.println("Binary Value is : " + binary);
      }
}
```

4. Array merging

```
import java.util.Scanner;
class MergeArray {
   public static void main(String[] args)
           {
      // declare variables
      int a1, b1, c1 = 0;
      // Scanner class object to read input values
      Scanner s = new Scanner(System.in);
      // read size of array from user
      System.out.print("Enter the size of first array : ");
      a1 = s.nextInt();
      int a[] = new int[a1];
      System.out.print("Enter the size of second array : ");
      b1 = s.nextInt();
      int b[] = new int[b1];
      c1 = a1 + b1;
      int[] c = new int[c1];
      // read array elements from user
      System.out.print("Enter elements of first array : ");
      for (int i = 0; i < a1; i++) {
         a[i] = s.nextInt();
      }
      System.out.print("Enter elements of second array : ");
      for (int i = 0; i < b1; i++) {
         b[i] = s.nextInt();
      }
      // merge two arrays
      for (int i = 0; i < a1; i = i + 1)
                     {
         c[i] = a[i];
      }
      for (int i = 0; i < b1; i = i + 1)
                     {
         c[a1 + i] = b[i];
```

```
      }
      // display merged array
      System.out.println("The merged array is : ");
      for (int i = 0; i < c1; i = i + 1)
                        {
         System.out.println(c[i]);
      }
   }
}
```

5. Trace and Transpose

```
import java.util.Scanner;
class Transpose {
   public static void main(String[] args) {
            int[][] matrix;
      int[][] transpose;
      int sum = 0, r, c;
            Scanner in = new Scanner(System.in);
            System.out.print("Enter the Number of Rows : ");
      r = in.nextInt();
      System.out.print("Enter the Number of Columns : ");
      c = in.nextInt();

                  matrix = new int[r][c];
         transpose = new int[c][r];
                  System.out.print("Enter the Elements of Matrix : ");
         for (int i = 0; i < r; i++) {
            for (int j = 0; j < c; j++) {
               matrix[i][j] = in.nextInt();
            }
         }                  System.out.println("\nGiven Matrix");
         for (int i = 0; i < r; i++) {
            for (int j = 0; j < c; j++) {
               System.out.print(matrix[i][j] + " ");
            }
            System.out.print("\n");
         }
                  for (int i = 0; i < r; i++) {
            for (int j = 0; j < c; j++) {
               transpose[j][i] = matrix[i][j];
               if (i == j) {
                  sum = sum + (matrix[i][j]);
               }
            }
         }
                  System.out.println("\nThe Trace of the Given Matrix is = " + sum);
         System.out.println("\nTranspose Matrix");
         for (int i = 0; i < c; i++) {
            for (int j = 0; j < r; j++) {
               System.out.print(transpose[i][j] + " ");
            }
            System.out.print("\n");
```

```
        }
      }
   }
```

6.Sum of digits and reverse

```
import java.util.Scanner;
class SumRev {
   // declare variables
   int a, m = 0, sum = 0;
   // function to find and display sum of digits
   void sum(int num) {
      do {
         a = num % 10;
         sum = sum + a;
         num = num / 10;
      } while (num > 0);
      System.out.println("Sum of digits = " + sum);
   }
   // function to find and display reverse
   void reverse(int num) {
      do {
         a = num % 10;
         m = m * 10 + a;
         num = num / 10;
      } while (num > 0);
      System.out.println("Reverse : " + m);
   }
}
class SumRevMain {
   public static void main(String[] args) {
      // declare variables
      int n;
      // Scanner class object to read input values
      Scanner sc = new Scanner(System.in);
      // read a number from user
      System.out.print("Enter any number : ");
      n = sc.nextInt();
      // create an object of class SumRev and call functions
      SumRev obj = new SumRev();
      obj.sum(n);
      obj.reverse(n);
   }
}
```

7.Number Sorting

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

      //Initialize array
      int [] arr = new int [] {5, 2, 8, 7, 1};
      int temp = 0;
      System.out.println("Elements of original array: ");
      for (int i = 0; i < arr.length; i++) {
         System.out.print(arr[i] + " ");
      }
      for (int i = 0; i < arr.length; i++) {
         for (int j = i+1; j < arr.length; j++) {
            if(arr[i] > arr[j]) {
               temp = arr[i];
               arr[i] = arr[j];
               arr[j] = temp;
            }
         }
      }

      System.out.println();
               System.out.println("Elements of array sorted in ascending order: ");
      for (int i = 0; i < arr.length; i++) {
         System.out.print(arr[i] + " ");
      }
   }
}
```

8.Addition of Complex numbers

```
class Complex {
   double real, img;

   // constructor to initialise complex number
   Complex(int r, int i) {
      real = r;
      img = i;
   }

   // function to add to complex numbers
   Complex addComp(Complex C1, Complex C2) {
      Complex sum = new Complex(0, 0);
      sum.real = C1.real + C2.real;
      sum.img = C1.img + C2.img;
      return sum;
   }
}

class ComplexMain {
   public static void main(String[] args) {
      // create 3 complex numbers using objects of Complex class
      Complex C1 = new Complex(2, 3);
```

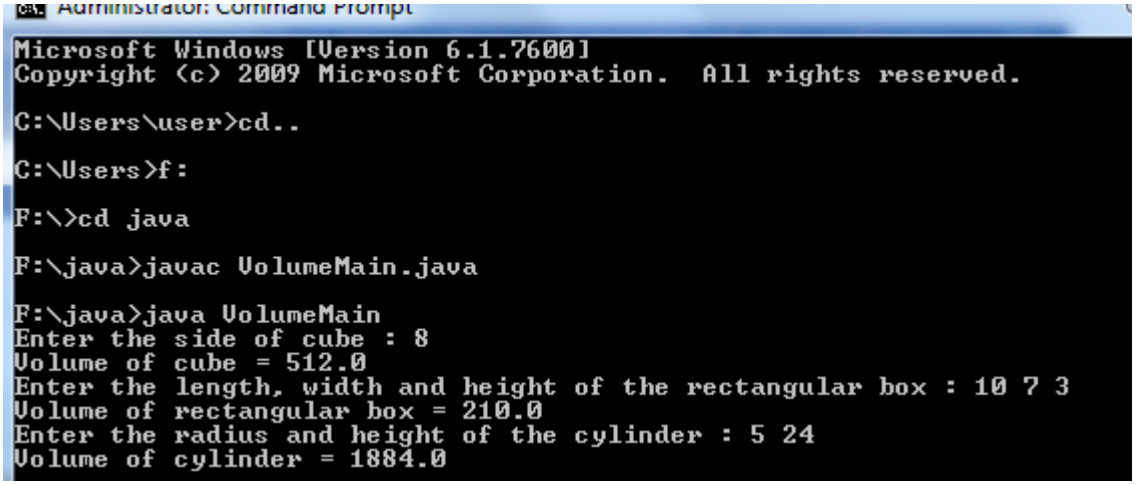
```
        Complex C2 = new Complex(5, 6);
        Complex C3 = new Complex(0, 0);

        // print two complex numbers
        System.out.println("Complex number 1 : " + C1.real + " + " + C1.img + "i");
        System.out.println("Complex number 2 : " + C2.real + " + " + C2.img + "i");

        // add 2 complex numbers and display it
        C3 = C3.addComp(C1, C2);
        System.out.println("Sum of complex numbers : " + C3.real + " + " + C3.img + "i");
    }
}
```

9.FUNCTION OVERLOADING

```
import java.util.Scanner;
class Volume {
    double vol, a, b, c;
    // volume of cube
    void findVolume(double a) {
        System.out.println("Volume of cube = " + (a * a * a));
    }
    // volume of rectangular box
    void findVolume(double a, double b, double c) {
        System.out.println("Volume of rectangular box = " + (a * b * c));
    }
    // volume of cylinder
    void findVolume(double a, double b) {
        System.out.println("Volume of cylinder = " + (3.14 * a * a * b));
    }
}
class VolumeMain {
    public static void main(String args[]) {
        double a, b, c;
        // Scanner class object to read input values
        Scanner s = new Scanner(System.in);
        // object of Volume class to call methods
        Volume obj = new Volume();
        // read sides / length,breadth / radius, height from user
        // call findVolume() with parameters
        System.out.print("Enter the side of cube : ");
        a = s.nextDouble();
        obj.findVolume(a);
        System.out.print("Enter the length, width and height of the rectangular box : ");
        a = s.nextDouble();
        b = s.nextDouble();
        c = s.nextDouble();
        obj.findVolume(a, b, c);
        System.out.print("Enter the radius and height of the cylinder : ");
        a = s.nextDouble();
        b = s.nextDouble();
        obj.findVolume(a, b);
    }
}
```



10. abstract class

```
import java.util.*;

abstract class Shape {
	int length, breadth;

	Scanner input = new Scanner(System.in);

	abstract void Area();

}

class Rectangle extends Shape {
	void Area() {
		System.out.println("*** Finding the Area of Rectangle ***");
		System.out.print("Enter length and breadth: ");
		length = input.nextInt();
		breadth = input.nextInt();
		System.out.println("The area of Rectangle is: " + length * breadth);
	}
}

class Triangle extends Shape {
	void Area() {
		System.out.println("\n*** Finding the Area of Triangle ***");
		System.out.print("Enter Base And Height: ");
		length = input.nextInt();
		breadth = input.nextInt();
		System.out.println("The area of Triangle is: " + (length * breadth) / 2);
	}
}

public class AbstractClass {
	public static void main(String[] args) {
		Rectangle rec = new Rectangle();
		rec.Area();

		Triangle tri = new Triangle();
		tri.Area();
```

```
        }
}
```

11.THREADS IN JAVA

```
import java.util.Scanner;
public abstract class OddEvenThread implements Runnable {
   public static void main(String[] args) {
      int limit;
      // Scanner class object to read input values
      Scanner sc = new Scanner(System.in);
      //read limit from user
      System.out.print("Enter the limit : ");
      limit = sc.nextInt();

      // create two threads
      Thread t1 = new Thread(new OddRunnable(limit));
      Thread t2 = new Thread(new EvenRunnable(limit));
      // Start both threads
      t1.start();
      t2.start();
   }
}
class OddRunnable implements Runnable {
   int limit;
   public OddRunnable(int limit) {
      this.limit = limit;
   }
   public void run() {
      for (int even =2;even <= limit;even+=2) {
         System.out.println("Even thread : " + even);
      }
   }
}
class EvenRunnable implements Runnable {
   int limit;
   public EvenRunnable(int limit) {
      this.limit = limit;
   }
   public void run() {
      for (int odd=1;odd <= limit;odd+=2) {
         System.out.println("Odd thread : " + odd);
      }
   }
}
```

12. Bank Account

```
import java.io.*;
import java.util.*;
import java.util.Scanner;
```

```
import java.util.Random;

class Bank
{
   public String nameOfDepositor;
   public int accNumber;
   public String accType;
   public double balanceAmount;

   public void assignValues(String nameOfDepositor, String accType,
double balanceAmount)
   {
      this.nameOfDepositor=nameOfDepositor;
      this.accType=accType;
      this.balanceAmount=balanceAmount;

      Random random = new Random();
      this.accNumber=random.nextInt(1000000);
      System.out.println("Your new account number is: "+accNumber);
   }
   public void depositAmount(double amount)
   {
      if(accNumber==0)
         System.out.println("!You don't have bank account to
deposit\nNote:please assign values to create an account");
      else
      {
         balanceAmount+=amount;
         System.out.println("Amount deposited successfully...");
      }
   }
   public void withdrawAmount(double amount)
   {
      if(accNumber==0)
```

```
        System.out.println("!You don't have bank account to
credit\nNote:please assign values to create an account");
      else if(balanceAmount>amount)
      {
        balanceAmount-=amount;
        System.out.println("Amount Debited successfully...");
      }
      else
        System.out.println("! Insufficient balance");
   }
   public void displayDetails()
   {
      if(accNumber==0)
        System.out.println("!You don't have bank account\nNote:please
assign values to create an account");
      else
      {
        System.out.println("Name of the Depositor:
"+nameOfDepositor);
        System.out.println("Balance amount in the account:
"+balanceAmount);
      }
   }
   public void getInput()
   {
      System.out.println("How can i help you?");
      System.out.println("1. Open account");
      System.out.println("2. Deposit amount");
      System.out.println("3. Withdraw amount");
      System.out.println("4. Account details");
      System.out.println("5. Exit");
      System.out.print("Please choose from above (Eg.2): ") ; }  }
class  Main
{
   public static void main(String[] s)  throws IOException
```

```
  {
     System.out.println("::::::::::WELCOME TO XYZ BANK::::::::::");
     Bank newAccount=new Bank();
     Scanner scan=new Scanner(System.in);
     boolean process=true;
     int continueState=0;
           while(continueState==0)
     {
        newAccount.getInput();
        int currentProcess=scan.nextInt();

        if(currentProcess==1)
        {
           System.out.print("Enter your name: ");
           String nameOfDepositor=scan.next();
           System.out.print("Enter your account type: ");
           String accType=scan.next();
           System.out.print("Enter your opening balance: ");
           double  balanceAmount=scan.nextDouble();
           newAccount.assignValues(nameOfDepositor, accType,
balanceAmount);
        }
        else if(currentProcess==2)
        {
           System.out.print("Enter amount to deposit: ");
           newAccount.depositAmount(scan.nextDouble());
        }
        else if(currentProcess==3)
        {
           System.out.print("Enter amount to withdraw: ");
           newAccount.withdrawAmount(scan.nextDouble());
        }
        else if(currentProcess==4)
        {
           newAccount.displayDetails();
```

```
        }
        else if(currentProcess==5)
        {
           continueState=1;
           System.out.println("THANK YOU");
        }

        System.out.print ("press 0 to continue... ");
        continueState=scan.nextInt();
     }
  }
}
```

13. Exception Handling

```
class Main {
  public static void main(String[] args) {
    try {
      // code that generates exception
      int divideByZero = 5 / 0;
    }
    catch (ArithmeticException e) {
      System.out.println("ArithmeticException => " +
e.getMessage());
    }
    finally {
      System.out.println("This is the finally block");
```

```
} } }
```

14. Applet

```
import java.applet.*;
import java.awt.*;
/*<applet code ="Smiley" width=600 height=600>
</applet>*/

public class Smiley extends Applet {
	public void paint(Graphics g)
	{

		// Oval for face outline
		g.drawOval(80, 70, 150, 150);
		// Ovals for eyes
```

```
        // with black color filled
        g.setColor(Color.BLACK);
        g.fillOval(120, 120, 15, 15);
        g.fillOval(170, 120, 15, 15);
        // Arc for the smile
        g.drawArc(130, 180, 50, 20, 180, 180);
    }
}

//javac Smiley.java
//appletviewer Smiley.java
```

15. Student class

```
import java.lang.*;
import java.io.*;
class student
{
String name;
int rno;
String course;
void getdata() throws IOException
{
BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
System.out.println("Enter name:");
name=br.readLine();
System.out.println("Enter rollno:");
rno=Integer.parseInt(br.readLine());
System.out.println("Enter the course :");
course=br.readLine();
}
void show()
{
System.out.println(" rollno:"+rno);
System.out.println(" name:"+name);
System.out.println(" course:"+course);
}
}
class mark extends student
```

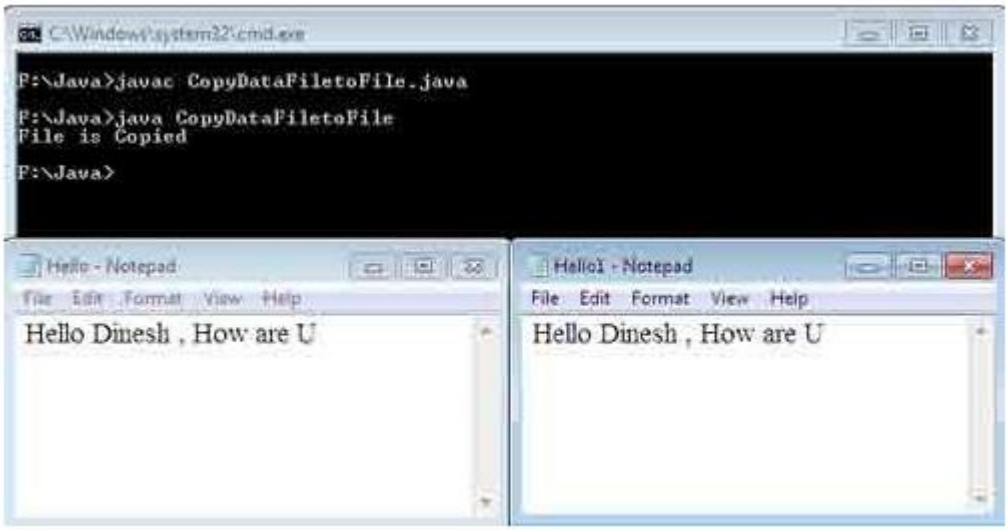
```
{
int []m= new int[5];
int tot=0;
float avg;
void getdata() throws IOException
{
super.getdata();
BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
System.out.println("Enter marks of five subjects:");
for(int i=0;i<5;i++)
{
m[i]=Integer.parseInt(br.readLine());
tot=tot+m[i];
}
avg=tot/5;
}


void show()
{
System.out.println("Details of student\n");
super.show();
System.out.println("\n marks of subjects\n");
for(int i=0;i<5;i++)
{
System.out.println("marks of sub["+(i+1)+"]:"+m[i]);
}
System.out.println(" total marks:"+tot);
System.out.println(" average mark:"+avg);
}
}
class sdetails
{
public static void main(String args[]) throws IOException
{
mark r=new mark();
r.getdata();
r.show();
}
}
```

16. file

```
import java.io.*;
 class CopyDataFiletoFile
 {
            public static void main(String args[])throws IOException
        {
             FileInputStream Fread =new FileInputStream("Hello.txt");
             FileOutputStream Fwrite=new FileOutputStream("Hello1.txt") ;
             System.out.println("File is Copied");
             int c;
             while((c=Fread.read())!=-1)
             Fwrite.write((char)c);
             Fread.close();
```

```
            Fwrite.close();
        }
}
```



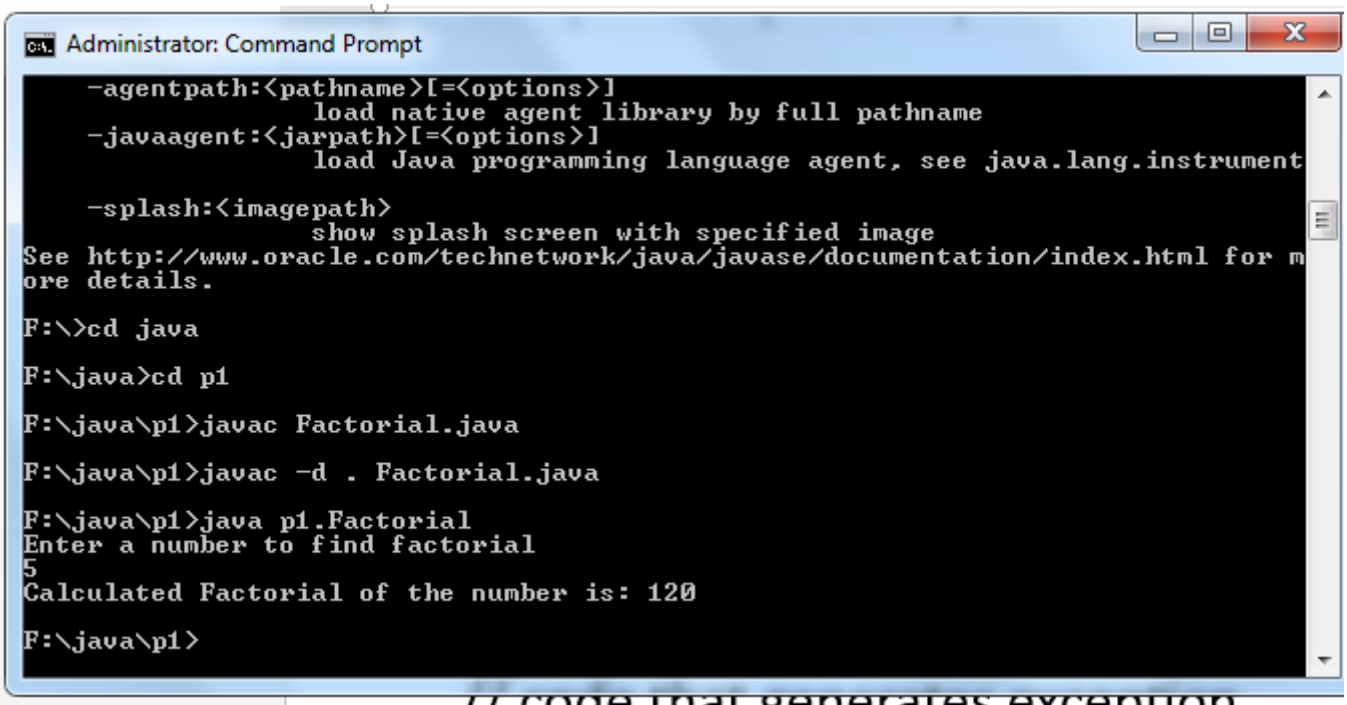
17. Factorial using package

```
package p1;
import java.util.Scanner;
import java.util.*;
public class Factorial
{
        static int fact(int a)
    {
          if(a <= 1)
            return 1;
          else
            return a * fact(a-1);
    }

  public static void main(String[] args)
  {
        System.out.println("Enter a number to find factorial");
        Scanner scan = new Scanner(System.in);
        int n = scan.nextInt();
        int Fact = fact(n);
        System.out.println("Calculated Factorial of the number is: " + Fact);

   }
```

}



// code that generates exception