**Q1)that takes a number as input and prints its multiplicatons table upto 10**

import java.util.Scanner;  
public class Pract1  
{  
 public static void main(String[] args)  
 {  
 Scanner st = new Scanner(System.*in*);  
 System.*out*.println("Enter number");  
 int num = st.nextInt();  
 for (int i = 1; i <= 10; i++)  
 {  
 System.*out*.println(num+"\*"+i+"="+num\*i);  
 }  
 }  
}

**Q2)to display the following pattern**

import java.util.Scanner;  
public class Pract2  
{  
 public static void main(String[] args)  
 {  
 int i,j;  
 for(i=5;i>=1;i--)  
 {  
 for(j=1;j<=i;j++)  
 {  
 System.*out*.print("\*");  
 }  
 System.*out*.println();  
 }  
 }  
}

**Q3)to print the area and perimeter of circle**

import java.util.Scanner;  
public class Pract3  
{  
 public static void main(String[] args)  
 {  
 double r=1.2f;  
 double perimeter=2\*3.14\*r;  
 double area=3.14\*r\*r;  
 System.*out*.println("Perimeter is: "+perimeter);  
 System.*out*.println("Area is: "+area);  
 }  
}

**Q4)to add two binary numbers**

import java.util.Scanner;  
public class Pract4  
{  
 public static void main(String[] args)  
 {  
 long b1,b2;  
 int i=0,rem=0;  
 int[] sum = new int[20];  
 Scanner in = new Scanner(System.*in*);  
 System.*out*.println("1st Binary number");  
 b1 = in.nextLong();  
 System.*out*.println("2nd Binary number");  
 b2 = in.nextLong();  
 while (b1 != 0 || b2 != 0)  
 {  
 sum[i++]=(int)((b1%10+b2%10+rem)%2);  
 rem=(int)((b1%10+b2%10+rem)%2);  
 b1=b1/10;  
 b2=b2/10;  
 }  
 if(rem!=0)  
 {  
 sum[i++]=rem;  
 }  
 --i;  
 System.*out*.println("Sum of b1+b2:=");  
 while(i>=0)  
 {  
 System.*out*.println(sum[i--]);  
 }  
 }  
}

**Q5)to convert a decimal number to binary number and vice versa**

import java.util.Scanner;  
class Pract5  
{  
 public static void main(String a[]) {  
 int num1;  
 String num2;  
 Scanner st = new Scanner(System.*in*);  
 System.*out*.print("Enter Binary number: ");  
 num2 = st.nextLine();  
 System.*out*.print("Enter Decimal number: ");  
 num1 = st.nextInt();  
 System.*out*.println("Binary to Decimal :" + Integer.*parseInt*(num2, 2));  
 System.*out*.println("Decimal to Binary." + Integer.*toBinaryString*(num1));  
 }  
}

**Q6)to reverse a string**

import java.util.\*;  
public class Pract6  
{  
 public static void main(String[] args) {  
 String s, t = "";  
 Scanner in = new Scanner(System.*in*);  
 System.*out*.println("Enter a string: ");  
 s = in.nextLine();  
 int length = s.length();  
 for (int i =length-1; i>=0; i--)  
 t=t+ s.charAt(i);  
 System.*out*.println("Reverse: "+t);  
 }  
}

**Q7)to count the letter, space ,numbers and other characters of an input string**

import java.util.Scanner;  
public class Pract7 {  
 public static void main(String[] args) {  
 String st;  
 Scanner in = new Scanner(System.*in*);  
 System.*out*.println("Enter a string: ");  
 st = in.nextLine();  
 *count*(st);  
 }  
  
 public static void count(String x)  
 {  
 char[] ch = x.toCharArray();  
 int l = 0, s = 0, n = 0, a = 0;  
 for (int i = 0; i < x.length(); i++)  
 {  
 if (Character.*isLetter*(ch[i]))  
 {  
 l++;  
 } else if (Character.*isDigit*(ch[i]))  
 {  
 n++;  
 } else if (Character.*isSpaceChar*(ch[i]))  
 {  
 s++;  
 } else  
 {  
 a++;  
 }  
 }  
 System.*out*.println("Letter:" + l);  
 System.*out*.println("Space: " + s);  
 System.*out*.println("Number:" + n);  
 System.*out*.println("Other: " + a);  
 }  
}

**Q8)that calculate the sum of digits from 0-9**

public class Pract8  
{  
 public static long calc()  
 {  
 long x=0L;  
 char c[] = {'0','1','2','3','4','5','6','7','8','9'};  
 long sum=0L;  
 for(int i=c.length-1;i>=0;i--)  
 {  
 x = c[i] - '0';  
 sum = sum + x;  
 }  
return sum;  
 }  
 public static void main(String[] args)  
 {  
 long r=*calc*();  
 System.*out*.println("Integer is " + r);  
 }  
}

**Q9) smallest and largest element from array**

public class Pract9  
{  
 public static void main(String[] args)  
 {  
 int num[] = new int[]{13,22, 27,11,4};  
 int s= num[0];  
 int g= num[0];  
 int i;  
 for(i=0; i<num.length; i++)  
 {  
 System.*out*.println(num[i]);  
 }  
 for(i=1; i<num.length; i++)  
 {  
 if(num[i] >g)  
 g=num[i];  
 else if (num[i] < s)  
 s= num[i];  
 }  
 System.*out*.println("Largest Number in array :" + g);  
 System.*out*.println("Smallest Number in array:" + s);  
 }  
}

**Q10) class sort data that contains the method asec() and desc()**

class SortData  
{  
 int n, temp, i, j;  
 public void desc(int num[])  
 {  
 for (i = 0; i < num.length; i++)  
 {  
 for (j = i + 1; j < num.length; j++)  
 {  
 if (num[i] < num[j])  
 {  
 temp = num[i];  
 num[i] = num[j];  
 num[j] = temp;  
 }  
 }  
 }  
 System.*out*.println("Descending Order:");  
 for (i = 0; i < num.length; i++)  
 {  
 System.*out*.print(num[i] + " ");  
 }  
 }  
 public void asec(int num1[])  
 {  
 for (i = 0; i < num1.length; i++)  
 {  
 for (j = i + 1; j < num1.length; j++)  
 {  
 if (num1[i] > num1[j])  
 {  
 temp = num1[i];  
 num1[i] = num1[j];  
 num1[j] = temp;  
 }  
 }  
 }  
 System.*out*.println("\nAscending Order:");  
 for(i=0;i<num1.length;i++)  
 {  
 System.*out*.print(num1[i] + " ");  
 }  
 }  
}  
public class Pract10  
{  
 public static void main(String[] args)  
 {  
 SortData s1=new SortData();  
 int ar[]=new int[]{13,22,27,11,4};  
 s1.desc(ar);  
 s1.asec(ar);  
 }  
}

**Q11)** **Designed a class that demonstrates the use of constructor and destructor.**

public class Pract11  
{  
 public Pract11()  
 {  
 System.*out*.println("Hello");  
 }  
 public void finalize()  
 {  
 System.*out*.println("Destroyed");  
 }  
  
 public static void main(String args[])   
 {  
 Pract11 s1 = new Pract11();  
 s1 = null;  
 System.*gc*();  
 }  
}

**Q12) Write a java program to demonstrate the implementation of abstract class.**

abstract class Calc  
{  
public abstract int sqr(int n1);  
public abstract int cube(int n1);  
public void show()  
{  
 System.*out*.println("Hello");  
}  
}  
class Pract12 extends Calc  
{  
public int sqr(int n1)  
{  
 return n1 \* n1;  
}  
public int cube(int n1)  
{  
 return n1 \* n1 \* n1;  
}  
public static void main(String args[])  
{  
 Pract12 st = new Pract12();  
 System.*out*.println(st.sqr(3));  
 System.*out*.println(st.cube(4));  
 st.show();  
}  
}

**Q13)** **A java program to implement single level inheritance.**

class Demo  
{  
 float pi = 3.14f;  
  
 void show()  
 {  
 System.*out*.println("Area of circle");  
 }  
}  
class Pract13 extends Demo  
{  
 float r = 2.0f;  
  
 void area() {  
 System.*out*.println(pi \* r \* r);  
 }  
  
 public static void main(String args[]) {  
 Pract13 p = new Pract13();  
 p.show();  
 p.area();  
 }  
}

**Q14) A java program to implement method overriding.**

class A  
{  
void show( )  
{  
 System.*out*.println("Base Class");  
}  
}  
 class B extends A  
 {  
 void show( )  
 {  
 System.*out*.println("Derieved Class");  
 }  
}  
class Pract14  
{  
 public static void main(String args[])  
 {  
 B s=new B();  
 s.show();  
 }  
}

**Q15) A java program to implement multiple inheritance.**

interface S  
{  
public void show();  
}  
interface T extends S  
{  
 public void display();  
}  
class Pract15 implements T  
{  
 public void show() {  
 System.*out*.println("From Interface S");  
 }  
  
 public void display() {  
 System.*out*.println("From Interface T");  
 }  
  
 public static void main(String args[]) {  
 Pract15 rl = new Pract15();  
 rl.show();  
 rl.display();  
 }  
}

**Q16)** **Design an AWT application to check whether the number entered in textbox is**

Even or not

import java.awt.\*;  
import java.awt.event.\*;  
public class EvenOddAWT extends Frame implements ActionListener  
{  
 Label label1, label2;  
 TextField textField;  
 Button button;  
  
 public EvenOddAWT()  
 {  
 setLayout(new FlowLayout());  
 label1 = new Label("Enter a number:");  
 textField = new TextField(10);  
 button = new Button("Check Even/Odd");  
 label2 = new Label("Result: ");  
 button.addActionListener(this);  
 add(label1);  
 add(textField);  
 add(button);  
 add(label2);  
 setSize(300, 150);  
 setTitle("Even or Odd Checker");  
 setVisible(true);  
 addWindowListener(new WindowAdapter()  
 {  
 public void windowClosing(WindowEvent e)  
 {  
 dispose();  
 }  
 });  
 }  
  
 public void actionPerformed(ActionEvent e)  
 {  
 try  
 {  
 int num = Integer.*parseInt*(textField.getText());  
 if (num % 2 == 0)  
 {  
 label2.setText("Result: Even");  
 } else  
 {  
 label2.setText("Result: Odd");  
 }  
 }  
 catch (NumberFormatException ex)  
 {  
 label2.setText("Invalid input! Enter a valid number.");  
 }  
 }  
 public static void main(String[] args)  
 {  
 new EvenOddAWT();  
 }  
}

Q17) Design an AWT application to calculate the factorial of a number

import java.awt.\*;  
import java.awt.event.\*;  
public class FactorialAWT extends Frame implements ActionListener  
{  
 Label label1, label2;  
 TextField textField;  
 Button button;  
 public FactorialAWT()  
 {  
 setLayout(new FlowLayout());  
 label1 = new Label("Enter a number:");  
 textField = new TextField(10);  
 button = new Button("Calculate Factorial");  
 label2 = new Label("Result: ");  
 button.addActionListener(this);  
 add(label1);  
 add(textField);  
 add(button);  
 add(label2);  
 setSize(300, 200);  
 setTitle("Factorial Calculator");  
 setVisible(true);  
 addWindowListener(new WindowAdapter()  
 {  
 public void windowClosing(WindowEvent e)  
 {  
 dispose();  
 }  
 });  
 }  
  
 public void actionPerformed(ActionEvent e)  
 {  
 try  
 {  
 int num = Integer.*parseInt*(textField.getText());  
 long fact = factorial(num);  
 label2.setText("Result: " + fact);  
 } catch (NumberFormatException ex)  
 {  
 label2.setText("Invalid input! Enter a number.");  
 }  
 }  
 public long factorial(int n)  
 {  
 long fact = 1;  
 for (int i = 1; i <= n; i++)  
 {  
 fact \*= i;  
 }  
 return fact;  
 }  
 public static void main(String[] args)  
 {  
 new FactorialAWT();  
 }  
}

**Q18) Design an AWT application to accept a number from user through textbox and**

print its reverse.

import java.awt.\*;  
import java.awt.event.\*;  
public class ReverseNumberAWT extends Frame implements ActionListener  
{  
 Label label1, label2;  
 TextField textField;  
 Button button;  
 public ReverseNumberAWT()  
 {  
 setLayout(new FlowLayout());  
 label1 = new Label("Enter a number:");  
 textField = new TextField(10);  
 button = new Button("Reverse Number");  
 label2 = new Label("Result: ");  
 button.addActionListener(this);  
 add(label1);  
 add(textField);  
 add(button);  
 add(label2);  
 setSize(300, 200);  
 setTitle("Reverse Number");  
 setVisible(true);  
 addWindowListener(new WindowAdapter()  
 {  
 public void windowClosing(WindowEvent e)  
 {  
 dispose();  
 }  
 });  
 }  
 public void actionPerformed(ActionEvent e)  
 {  
 try  
 {  
 int num = Integer.*parseInt*(textField.getText());  
 int reversedNum = reverseNumber(num);  
 label2.setText("Result: " + reversedNum);  
 } catch (NumberFormatException ex)  
 {  
 label2.setText("Invalid input! Enter a number.");  
 }  
 }  
 public int reverseNumber(int n)  
 {  
 int reverse = 0;  
 while (n != 0)  
 {  
 int digit = n % 10;  
 reverse = reverse \* 10 + digit;  
 n /= 10;  
 }  
 return reverse;  
 }  
  
 public static void main(String[] args) {  
 new ReverseNumberAWT();  
 }  
}

**Q19) Write a Java AWT program to accept a number from user (TextBox) and display**

the cube of the number

import java.awt.\*;  
import java.awt.event.\*;  
public class CubeCalculatorAWT extends Frame implements ActionListener  
{  
 Label label1, label2;  
 TextField textField;  
 Button button;  
  
 public CubeCalculatorAWT()  
 {  
 setLayout(new FlowLayout());  
 label1 = new Label("Enter a number:");  
 textField = new TextField(10);  
 button = new Button("Calculate Cube");  
 label2 = new Label("Result: ");  
 button.addActionListener(this);  
 add(label1);  
 add(textField);  
 add(button);  
 add(label2);  
 setSize(300, 200);  
 setTitle("Cube Calculator");  
 setVisible(true);  
 addWindowListener(new WindowAdapter()  
 {  
 public void windowClosing(WindowEvent e)  
 {  
 dispose();  
 }  
 });  
 }  
 public void actionPerformed(ActionEvent e)  
 {  
 try  
 {  
 int num = Integer.*parseInt*(textField.getText());  
 int cube = num \* num \* num;  
 label2.setText("Result: " + cube);  
 }  
 catch (NumberFormatException ex)  
 {  
 label2.setText("Invalid input! Enter a number.");  
 }  
 }  
  
 public static void main(String[] args)  
 {  
 new CubeCalculatorAWT();  
 }  
}