Q1) multiplication table of 10

import java.util.Scanner;

public class pract1 {

public static void main(String[] args) {

Scanner st = new Scanner(System.in);

System.out.print("Enter number: ");

int num = st.nextInt();

for (int i = 1; i <= 10; i++) {

System.out.println(num + " \* " + i + " = " + num \* i);

}

}

}

Q2) star pattern

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) area and perimeter

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) add 2 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.print("1st binary number: ");

b1 = in.nextLong();

System.out.print("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.print("Sum of b1 + b2 =: ");

while (i >= 0) {

System.out.print(sum[i--]);

}

}

}

Q5) convert decimal to binary 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) 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) count letters,spaces, numbers and others

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) digits 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--) {

/\*the character '0' is coded as ASCII 48. So you need to

in the range 0..9\*/

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 number

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) sort data 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)construstor 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) 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)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)method overloading

class A {

void show() {

System.out.println("Base Class");

}

}

class B extends A {

void show() {

System.out.println("Derived Class");

}

}

class Pract14 {

public static void main(String args[]) {

B s = new B();

s.show();

}

}

Q15)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 r1 = new Pract15();

r1.show();

r1.display();

}

}

Q16)package

// Code of package

package mathematics;

public class Mathmethods {

public static float sqr(float n) {

return (n \* n);

}

public static int sqr(int n) {

return (n \* n);

}

public static double sqr(double n) {

return (n \* n);

}

public static long sqr(long n) {

return (n \* n);

}

}

import java.io.\*;

import mathematics.Mathmethods;

class sqr {

public static void main(String S[]) {

int a;

a = 2;

Mathmethods mm = new Mathmethods();

int b = Mathmethods.sqr(a);

System.out.println("Square is " + b);

}

}

Q17) add 2 matrices

class Pract16 {

public static void main(String args[]) {

int a[][] = {{1, 2}, {3, 4}};

int b[][] = {{5, 6}, {7, 8}};

int c[][] = new int[2][2];

int i, j;

System.out.println("Matrix A");

for (i = 0; i < 2; i++) {

for (j = 0; j < 2; j++) {

System.out.print(a[i][j] + " ");

}

System.out.println();

}

System.out.println("Matrix B");

for (i = 0; i < 2; i++) {

for (j = 0; j < 2; j++) {

System.out.print(b[i][j] + " ");

}

System.out.println();

}

}

}

System.out.println("Matrix A + B");

for(int i=0; i<2; i++) {

for(int j=0; j<2; j++) {

int c[i][j]= a[i][j] + b[i][j];

System.out.print(c + " ");

}

System.out.println();

}

}

}

Q18) AWT-textfield and textarea

import java.awt.\*;

import java.awt.event.\*;

public class AWTTextExample {

public static void main(String args[])

Frame frame = new Frame("AWT Text and TextArea Example");

Label label1 = new Label("Enter your name:");

TextField textField = new TextField("Default Name");

Label label2 = new Label("Enter comments:");

TextArea textArea = new TextArea("Default comment...", 5, 20); // 5 rows, 20 columns

frame.add(label1);

frame.add(textField);

frame.add(label2);

frame.add(textArea);

frame.setLayout(new FlowLayout());

frame.setSize(300, 300);

frame.addWindowListener(new WindowAdapter() {

public void windowClosing(WindowEvent we) {

System.exit(0);

}

});

frame.setVisible(true);

}

}

Q19) create a scrollbar (AWT)

import java.awt.\*;

public class SimpleScrollbarExample {

public static void main(String[] args) {

Frame frame = new Frame("Simple Scrollbar Example");

Scrollbar scrollbar = new Scrollbar(); // default vertical scrollbar

scrollbar.setBounds(50, 50, 20, 200); // position (x, y), width, height

frame.add(scrollbar);

frame.setLayout(null);

frame.setSize(300, 300);

frame.setVisible(true);

frame.addWindowListener(new java.awt.event.WindowAdapter() {

public void windowClosing(java.awt.event.WindowEvent we) {

System.exit(0);

}

});

}

Q20) checkbox class

import java.awt.\*;

public class CheckboxExample {

public static void main(String[] args) {

Frame f = new Frame("Checkbox Example");

// Creating checkbox

Checkbox checkbox = new Checkbox("Accept Terms");

// Add checkbox to the frame

f.add(checkbox);

// Set layout and size

f.setLayout(new FlowLayout());

f.setSize(300, 100);

f.setVisible(true);

}

}

Q21)list class

import java.awt.\*;

import java.awt.event.\*;

public class ListExample {

public static void main(String[] args) {

Frame f = new Frame("List Example");

// Create a List with multiple selection mode

List list = new List(5, true); // 5 visible rows, multiple selection enabled

// Add items to the list

list.add("Java");

list.add("Python");

list.add("C++");

list.add("JavaScript");

list.add("Ruby");

// Add an ItemListener to respond to selection

list.addItemListener(new ItemListener() {

public void itemStateChanged(ItemEvent e) {

System.out.println("Selected Item: " + list.getSelectedItem());

}

});

// Add the List to the frame

f.add(list);

// Set the layout and size

f.setLayout(new FlowLayout());

f.setSize(300, 200);

f.setVisible(true);

// Close the frame on close

f.addWindowListener(new WindowAdapter() {

public void windowClosing(WindowEvent we) {

System.exit(0);

}

});

}

}

Q22) label control

import java.awt.\*;

public class LabelExample {

public static void main(String[] args) {

Frame f = new Frame("Label Example");

// Create a Label with text and alignment

Label label = new Label("Hello, Java!", Label.CENTER);

label.setForeground(Color.BLUE);

// Add the label to the frame

f.add(label);

// Set layout and size

f.setLayout(new FlowLayout());

f.setSize(300, 100);

f.setVisible(true);

// Close the frame on window close

f.addWindowListener(new java.awt.event.WindowAdapter() {

public void windowClosing(java.awt.event.WindowEvent we) {

System.exit(0);

}

});

}

}