

Nombre: Giancarla Lizcano Di Costanzo María Rebeca Martínez Ferreira		Matrícula: 2682574 2867026
Nombre del curso: Fundamentos de programación	Nombre del profesor: Juan Miguel González García	
Módulo: 3	Actividad: 11, 12, 13, 14, 15	
Fecha: 7 de noviembre del 2017		
Bibliografía: Miscursos.tecmilenio.mx. (2017). Blackboard Learn. [online] Available at: https://miscursos.tecmilenio.mx/webapps/portal/execute/tabs/tabAction?tab_tab_group_id=_1_1 [Accessed 10 Nov. 2017].		

USO DE ARRAY LIST:

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package usodearraylist;

import java.util.ArrayList;
import javax.swing.JOptionPane;

/**
 *
 * @author Alumno
 */
public class UsoDeArrayList {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // Introducir Arreglo.
        ArrayList<String> nombres = new ArrayList();
        // Definir nombre por medio de variables.
        String elNombre;
        // Variable para seguir desarrollando nombres y para detener la
        introduccion de estos.
        int contador;
        // Variable para desplegar nombres.
        String losNombres = "";
        // Introducir Nombres y Guardarlos dentro del Arreglo.
        do {
            // Introducir un Nombre.
            elNombre = JOptionPane.showInputDialog(null, "Escribir un Nombre
que su inicial sea Mayuscula");
            nombres.add(elNombre);
            // Preguntar si se quiere seguir introduciendo mas nombres.
            contador = JOptionPane.showConfirmDialog(null, "¿Desea otro
Nombre?", "Arreglo de Nombres", JOptionPane.YES_NO_OPTION);
            // comprueba SI/NO.
        } while (contador == JOptionPane.YES_NO_OPTION);

        // Desplegar los nombres del Arreglo.
        for (int i = 0; i < nombres.size(); i++) {

            losNombres += "Nombre: " + i + ", " + nombres.get(i) + "\n";
        }
    }
}
```

```
}  
  
JOptionPane.showMessageDialog(null, losNombres);  
  
}  
  
}
```

PSEUDOCÓDIGO:

INICIO
INTRODUCIR un nombre
NOMBRE
PREGUNTA ¿desea ingresar mas nombres?
YES_NO
YES introducir más nombres
NO mostrar los nombres
FIN

METODOS DE AREA Y PERIMETRO:

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package Metodosareayperimetro;

import java.util.Scanner;

/**
 *
 * @author Alumno
 */
public class Metodosareayperimetro {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // Definir metodos.
        // Introducir las variables.
        double base, altura, lado1Ti, lado2Ti;
        int num, num2 = 0;
        Scanner teclado = new Scanner (System.in);
        //Menú
        /*
        *Triangulo Equilatero, isosceles o escaleno
        *Triangulo Rectangulo
        */
        System.out.println("Seleccionar una opción");
        System.out.println("1.Triángulo");
        System.out.println("2.Triangulo Rectángulo");
        //Selección de opción.
        num = teclado.nextInt();
        switch (num) {
            case 1:
                System.out.println("Seleccionar el tipo de triángulo");
                System.out.println("1.Equilatero");
                System.out.println("2.Isosceles");
                System.out.println("3.Escaleno");
                switch (num2) {
                    case 1:
                        System.out.println("Introducir la base");
                        base = teclado.nextInt();
                        System.out.println("Introducir la altura");
                        altura = teclado.nextInt();
                        //realizar operación
                        String mensaje;
```

```
        mensaje = "El area es: " + doAreaTi(base,altura) + "\n";
        mensaje = "El perimetro es: " + doPeriTiEqui(base);
        System.out.println(mensaje);
        break;
    case 2:
        System.out.println("Introducir la base");
        base = teclado.nextInt();
        System.out.println("Introducir la altura");
        altura = teclado.nextInt();
        System.out.println("Introducir lado");
        ladoTi = teclado.nextInt();
        //Realizar operación
        mensaje = "El area es: " + doAreaTi(base,altura) + "\n";
        mensaje = "El perimetro es: " + doPeriTilsos (base, ladoTi);
        System.out.println(mensaje);
        break;
    case 3:
        System.out.println("Introducir la base");
        base = teclado.nextInt();
        System.out.println("Introducir la altura");
        altura = teclado.nextInt();
        System.out.println("Introducir lado");
        ladoTi = teclado.nextInt();
        System.out.println("Introducir segundo lado");
        lado2Ti = teclado.nextInt();
        //Realizar operacion
        mensaje = "El area es: " + doAreaTi(base,altura) + "\n";
        mensaje = "El perimetro es: " + doPeriTiEsca (base, ladoTi,
lado2Ti);

        System.out.println(mensaje);
        break;
    default:
        System.out.println("Opcion invalida");
        break;
    }
    break;
case 2:
    System.out.println("Introducir la base");
    base = teclado.nextInt();
    System.out.println("Introducir la altura");
    altura = teclado.nextInt();
    //realizar operacion.
    String mensaje;
    mensaje = "El area es: " + doAreaRe(base,altura) + "\n";
    mensaje = "El perimetro es: " + doPeriRe(base,altura);
    System.out.println(mensaje);
    break;
default:
    System.out.println("Opcion invalida");
    break;
```

```
}  
}  
//Metodo calculando area triangulo a= b* h/2  
public static double doAreaTi (double base,double altura){  
    return (base * altura) /2;  
}  
//Metodo calculando perimetro triangulo equilatero  
public static double doPeriTIEqui (double base){  
    return base * 3;  
}  
//Metodo calculando perimetro triangulo isoseles  
public static double doPeriTIsos (double base, double ladoTi){  
    return 2 * ladoTi + base ;  
}  
//Metodo calculando perimetro triangulo escaleno  
public static double doPeriTIESca (double base, double ladoTi, double  
lado2Ti){  
    return base + ladoTi + lado2Ti;  
}  
//Metodo calculando area triangulo rectangulo  
public static double doAreaRe (double base, double altura){  
    return base * altura;  
}  
//Metodo calculando perimetro del triangulo rectangulo  
public static double doPeriRe (double base, double altura){  
    return (base + altura) * 2;  
}  
}
```

PSEUDOCÓDIGO:

INICIO

ELECCIÓN de triángulos, Escaleno, Isósceles, Equilátero

INTRODUCCIÓN DE DATOS

Base

Altura

L1

L2

REALIZAR OPERACIÓN PARA LA OBTENCIÓN DEL ÁREA

b x a

DESPLEGAR MENSAJE EL AREA ES

MOSTRAR RESULTADO

FIN

CUADROS DE DIALOGO:

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package cuadrosdialogo;
import javax.swing.JOptionPane;

/**
 *
 * @author Alumno
 */
public class CuadrosDialogo {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // Identificar variables de entrada.
        String direccion ;
        direccion = JOptionPane.showInputDialog("Introducir datos", "introducir
direccion");
        JOptionPane.showMessageDialog(null, "Tu direccion es: " + direccion );
        // Establecer los valores.
        int numerodeCasa;
        numero = Integer.parseInt(JOptionPane.showInputDialog("Introducir
números","introducir numero entero"));
        JOptionPane.showMessageDialog(null, "Tu numero de casa es : " +
numero);
    }

}
```

PSEUDOCODIGO:

INICIO
ENTRADA escribe algo
MOSTRAR LO QUE SE ESCRIBIO
FIN

PRUEBA OBJETOS:

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package pruebaobjetos;

import ClasesGenericas.TPersona;

/**
 *
 * @author Alumno
 */
public class Pruebaobjetos {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // Definir persona.
        TPersona persona = new TPersona();
        // Proporcionar información para identificación de la persona.
        persona.setNombrePersona("Ana");
        persona.setCorreoElectronico("Ana.perez@gmail.com");
        persona.setTelefonoCelular("22995823050");
        // Mostrar información.
        System.out.println("Nombre Persona: " + persona.getNombrePersona());
        System.out.println("Correo Electronico: " +
persona.getCorreoElectronico());
        System.out.println("Celular: " + persona.getTelefonoCelular());
        //Definir otra persona
        TPersona otrapersona = new TPersona ("Juanito Caminador",
"juanito@gmail.com", "2291558899");

        System.out.println("Nombre persona: " +
otrapersona.getNombrePersona());
        System.out.println("Correo Electronico: " +
otrapersona.getCorreoElectronico());
        System.out.println("Celular: " + otrapersona.getTelefonoCelular());
    }
}
```


PSEUDOCODIGO:

INICIO

PROCESO introduce un nombre

ENTRADA Tpersona

PROCESO introduce un correo electrónico

ENTRADA correo electrónico

PROCESO introduce un número celular

ENTRADA numero celular

LOS DATOS SON “ + Tpersona + correoelectronico + telefonocelular”

FIN

TPERSONA:

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package ClasesGenericas;

/**
 *
 * @author Alumno
 */
public class TPersona {

    private String nombrePersona;
    private String correoElectronico;
    private String telefonoCelular;

    //Metodo constructor
    public TPersona () {

    }

    public TPersona(String nombrePersona, String correoElectronico, String
telefonoCelular) {
        this.nombrePersona = nombrePersona;
        this.correoElectronico = correoElectronico;
        this.telefonoCelular = telefonoCelular;
    }

    public String getNombrePersona() {
        return nombrePersona;
    }

    public void setNombrePersona(String nombrePersona) {
        this.nombrePersona = nombrePersona;
    }

    public String getCorreoElectronico() {
        return correoElectronico;
    }

    public void setCorreoElectronico(String correoElectronico) {
        this.correoElectronico = correoElectronico;
    }

    public String getTelefonoCelular() {
        return telefonoCelular;
    }
}
```

```
}  
  
public void setTelefonoCelular(String telefonoCelular) {  
    this.telefonoCelular = telefonoCelular;  
}  
  
}
```

Nombre: María Rebeca Martínez Ferreira. Giancarla Lizcano Di Costanzo.		Matrícula: al02867026 al02682574
Nombre del curso: Fundamentos de programación.	Nombre del profesor: Juan Miguel Gonzáles García.	
Módulo: 3	Actividad: 11 Tarea: 11	
Fecha: 02-Noviembre-2017		
Bibliografía: Serprogramadores. (2014). Mensajes de dialogo en Java con JOptionPane . 03 de Noviembre del 2017, de Serprogramadores Sitio web: https://serprogramador.es/programando-mensajes-de-dialogo-en-java-parte-1/ Jorge Villarreal. (s.f.). Cuadros de diálogo. 03 de Noviembre del 2017, de Programación Básica JAVA Sitio web: https://sites.google.com/site/programacionbasicajava/cuadros-de-dialogo		

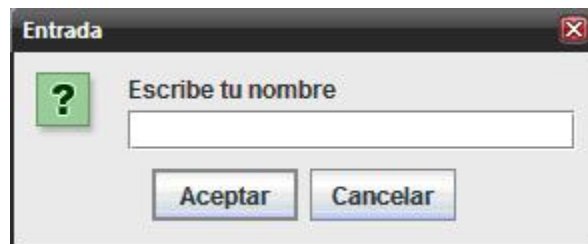
CUADROS DE DIÁLOGO

Los cuadros de diálogo son ventanas que están a la vista con objetivo de crear una interacción entre el usuario y el programa, este contiene información importante para la persona que lo esté utilizando y así informar sobre decisiones a tomar o errores que se presentaron al momento de realizar una acción.

Existen 4 métodos para realizarlos

- ❖ showMessageDialog
- ❖ showInputDialog
- ❖ showConfirmDialog
- ❖ showOptionDialog

Este es el aspecto de una ventana de diálogo:



Nombre: María Rebeca Martínez Ferreira. Giancarla Lizcano Di Costanzo.		Matrícula: al02867026 al02682574
Nombre del curso: Fundamentos de programación.	Nombre del profesor: Juan Miguel Gonzáles García.	
Módulo: 3	Actividad: 12 Tarea: 12	
Fecha: 03-Noviembre-2017		
Bibliografía: Programarya. (2013). Métodos en Java, funciones y procedimientos. Cómo hacerlos y usarlos. 03 de Noviembre del 2017, de Programarya Sitio web: https://www.programarya.com/Cursos/Java/Funciones		

MÉTODOS

Un método consta de una serie de instrucciones para la realización de una función específica. Estos son muy útiles para agilizar el programa pues en uno método se pueden almacenar datos que se repiten y en lugar de poner cada uno se guardan en este para ahorrar memoria, además de poder repetir datos, etc.

Estos son los métodos existentes:

- `Math.pow()`
- `Math.sqrt()`
- `Character.isDigit()`
- `System.out.println();`

Nombre: María Rebeca Martínez Ferreira. Giancarla Lizcano Di Costanzo.		Matrícula: al02867026 al02682574
Nombre del curso: Fundamentos de programación.	Nombre del profesor: Juan Miguel Gonzáles García.	
Módulo: 3	Actividad: 13 Tarea: 13	
Fecha: 03-Noviembre-2017		
Bibliografía: Oracle. (2017). Class ArrayList. 03 de Noviembre del 2017, de Oracle Sitio web: https://docs.oracle.com/javase/7/docs/api/java/util/ArrayList.html		

ARRAYLIST

Un ArrayList es una clase la cual nos permite poder almacenar datos que usualmente ocuparían mucha memoria, la ventaja que tienen los arrays es que su memoria es ilimitada y por lo tanto guardar datos para volver más rápido el programa no será un problema.

Clases de ArrayList:

`java.lang.Object`

`java.util.AbstractCollection()`

`java.util.AbstractList()`

`java.util.ArrayList()`

Nombre: María Rebeca Martínez Ferreira. Giancarla Lizcano Di Costanzo.		Matrícula: al02867026 al02682574
Nombre del curso: Fundamentos de programación.	Nombre del profesor: Juan Miguel González García.	
Módulo: 3	Actividad: 14 Tarea: 14	
Fecha: 03-Noviembre-2017		
Bibliografía:		

CLASES Y OBJETOS

Una clase define las características definitivas tales como nombre y el tipo que va a ser. Por su parte un objeto contiene los atributos que va a tener, si va a ser grande o chico, si es verde o naranja etc.

Para que un objeto funcione siempre debe ir contenido en una clase y que de ese modo cumpla con las instrucciones proporcionadas por la clase.

Nombre: María Rebeca Martínez Ferreira. Giancarla Lizcano Di Costanzo.		Matrícula: al02867026 al02682574
Nombre del curso: Fundamentos de programación.	Nombre del profesor: Juan Miguel Gonzáles García.	
Módulo: 3	Actividad: 15 Tarea: 15	
Fecha: 03-Noviembre-2017		
Bibliografía: JaGonzales. (2017). Crear Ventana En Java. 03 de Noviembre del 2017, de JaGonzales.org Sitio web: http://jagonzalez.org/crear-ventana-en-java/		

VENTANAS (JFRAME)

Las ventanas jframe tienen la misma función que un cuadro de diálogo con la diferencia de un atractivo visual mayor para el usuario, dándonos así ventanas más estilizadas.