

Nombre de la práctica	EXCEPCIONES (UNIDAD 5)			No.	1
Asignatura:	PROGRAMACION ORIENTADA A OBJETOS	Carrera:	INGENIERÍA EN SISTEMAS COMPUTACIONALES	Duración de la práctica (Hrs)	5 horas

**NOMBRE DEL ALUMNO:** OMAR ANAYA MARTINEZ

**GRUPO:** 3202

## I. Competencia(s) específica(s):

Conoce y aplica un lenguaje de programación para la resolución de problemas.

**Encuadre con CACEI:** Registra el (los) atributo(s) de egreso y los criterios de desempeño que se evaluarán en esta práctica.

No. atributo	Atributos de egreso del PE que impactan en la asignatura	No. Criterio	Criterios de desempeño	No. Indicador	Indicadores
1	El estudiante identificará los principios de las ciencias básicas para la resolución de problemas prácticos de ingeniería	CD1	Identifica problemas relacionados con aplicación de la ingeniería	I1	Análisis de problemas y/o necesidades
				I2	Empleo herramientas para el análisis
		CD2	Propone alternativas de solución	I1	Diseño algorítmico
				I3	Modelado de programas
				I4	Uso de metodologías
		CD3	Analiza y comprueba los resultados generados	I1	Comprobación de resultados
I2	Toma de decisiones				
3	El estudiante plantea soluciones basadas en tecnologías empleando su juicio ingenieril para valorar necesidades, recursos y resultados esperados.	CD1	Emplea los conocimientos adquiridos para el desarrollar soluciones	I1	Elección de metodologías, técnicas y/o herramientas para el desarrollo de soluciones
				I2	Uso de metodologías adecuadas para el desarrollo de proyectos
				I3	Generación de productos y/o proyectos
		CD2	Analiza y comprueba resultados	I1	Realizar pruebas a los productos obtenidos
				I2	Documentar información de las pruebas realizadas y los resultados

## II. Lugar de realización de la práctica (laboratorio, taller, aula u otro):

Laboratorio de cómputo y equipo de cómputo personal.

## III. Material empleado:

- Equipo de cómputo
- Netbeans - jdk

## IV. Desarrollo de la práctica:

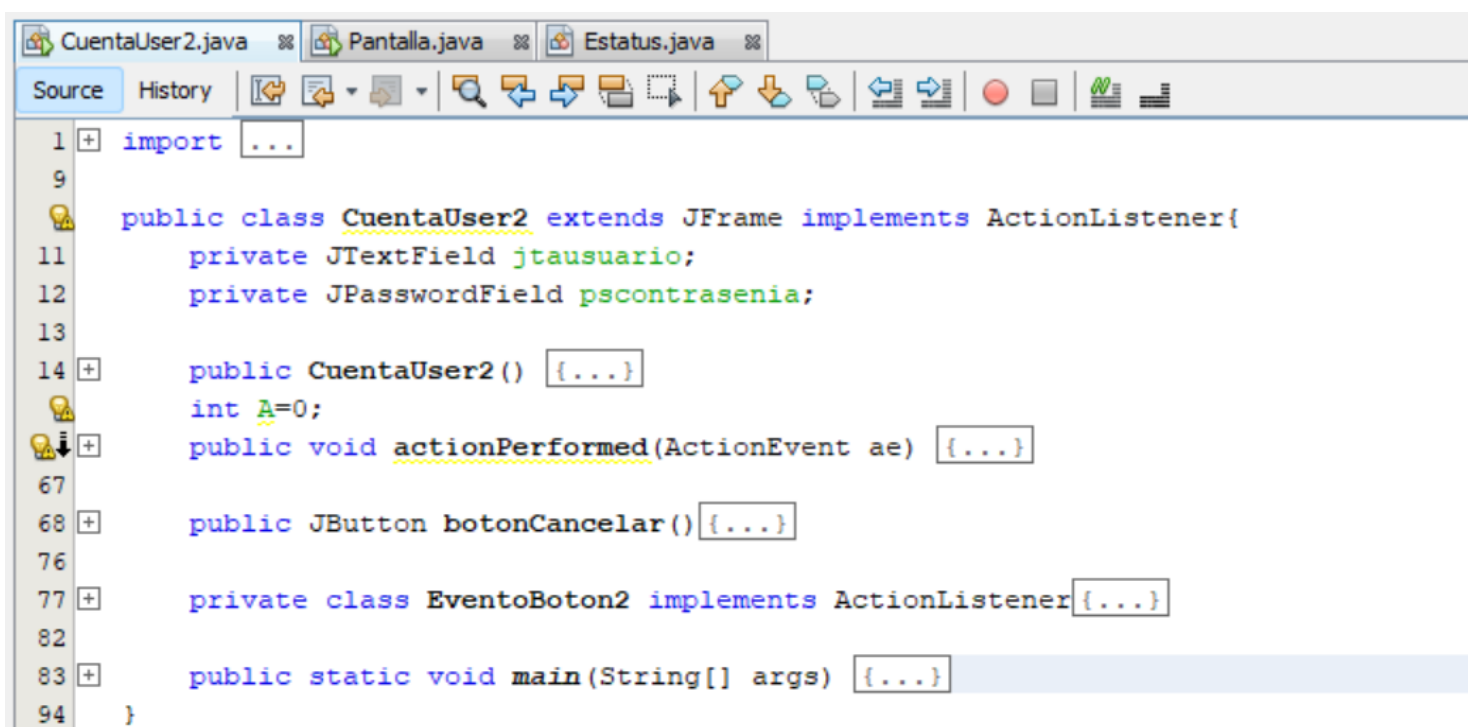
### UNIDAD 2

#### PROGRAMA 1:

##### A) Descripción del problema:

Con el programa realizado de herencia, incorporar alguna plantilla con el conocimiento visto por parte del tema polimorfismo.

##### B) Código de java



```
CuentaUser2.java  Pantalla.java  Estatus.java
Source  History
1  import ...
9
10 public class CuentaUser2 extends JFrame implements ActionListener{
11     private JTextField jtausuario;
12     private JPasswordField pscontrasenia;
13
14     public CuentaUser2() {...}
15     int A=0;
16     public void actionPerformed(ActionEvent ae) {...}
17
18     public JButton botonCancelar() {...}
19
20     private class EventoBoton2 implements ActionListener {...}
21
22     public static void main(String[] args) {...}
23 }
```



```

+ import ...
public class Pantalla extends JFrame implements ActionListener, Estatus{
    ImageFondo ejemplo = new ImageFondo();
    private JFrame ventProm;
    private JLabel tf, u1, u2, u3, u4, u5, rep, aprob, snCal;
    //private JLabel unds = new JLabel();
    private JComboBox semestres, carreras,sems;
    private JPanel p1, p2, p3, p4, p5, panRep, panApr;
    private JTextField[] cubito1,cubito2,cubito3,cubito4,cubito5;
    int f,g,h,j,i,C,p,q;

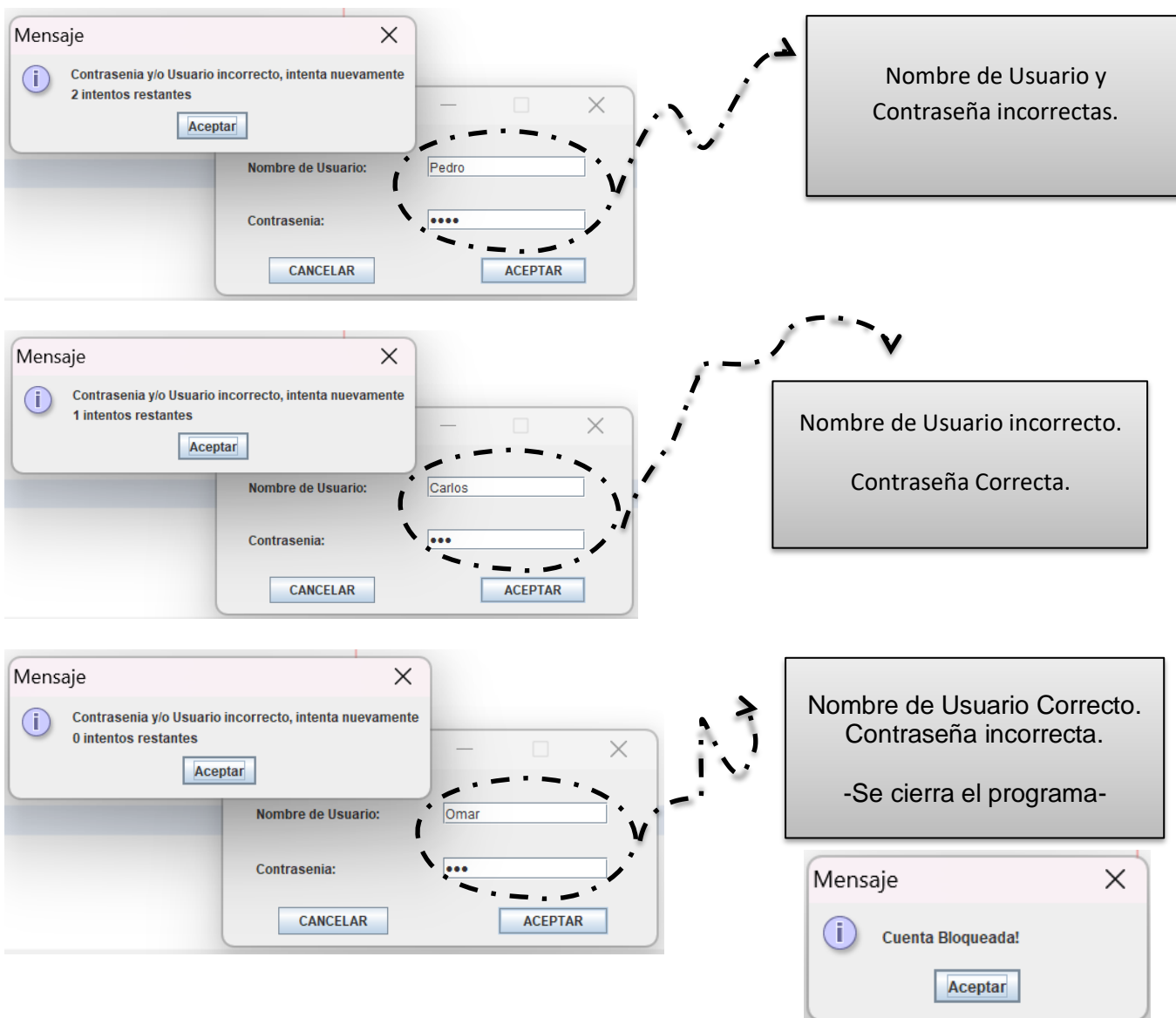
+ private void primerSem() {...}
+ private void segundoSem() {...}
+ private void tercerSem() {...}
+ private void cuartoSem() {...}
+ private void quintoSem() {...}
    int J, h1, h2, h3, h4, h5;
+ public Pantalla() {...}
+ public class ImageFondo extends JPanel {...}

+ public JButton botonLimp() {...}
+ private class EvBotnLimp implements ActionListener {...}
+ public JButton botonCancelar() {...}
+ private class EvBotnCancel implements ActionListener {...}
+ public JButton botonProm() {...}
    int ret = 0;
    String sem;
+ private void mostrarSemestres() {...}
    byte P1, P2, P3, P4, P5;
    int U1, U2, U3, U4, U5, g1,g2,g3,g4,g5, gt, gtl;
+ public void actionPerformed(ActionEvent ae) {...}
    byte op, fx;
    String calif;
+ private class EvBotnProm implements ActionListener {...}
    @Override
+ public void evCalif() {...}
    int f1, f2,f3,f4,f5;
    @Override
+ public void premio() {...}
+ public static void main(String [] args)throws IOException {...}
}

```

```
CuentaUser2.java Pantalla.java Estatus.java
Source History
public interface Estatus{
    public void evCalif();
    public void premio();
}
4
5
```

## C) Pantallas resultantes-probando con distintos datos.





Nombre de Usuario: Omar

Contraseña: ....

CANCELAR ACEPTAR

Nombre de Usuario y  
Contraseñas correctos.  
Ingresa al apartado de  
semestres.

CARRERA: Ing. Logística

SEMESTRE: Ter Semestre

GUARDAR CANCELAR LIMPIAR PROMEDIO

```
43 String A1, A21;
44 try{
45     int A2;
46     A1 = jtausuario.getText();
47     A21 = pscontrasenia.getText();
48     A2 = Integer.parseInt(A21);
49     A += 1;
50     if ("Omar".equals(A1)){
51         if (A2 == 123){
52             Pantalla pantalla = new Pantalla();
53             dispose();
54             A=0;
55         }else{
56             JOptionPane.showMessageDialog(null,"Contraseña y/o Usuario incorrecto, intenta nuevamente\n"+(3-A)+" intentos restantes");
57             jtausuario.setText(null);
58             pscontrasenia.setText(null);
59         }
60     }else{
61         JOptionPane.showMessageDialog(null,"Contraseña y/o Usuario incorrecto, intenta nuevamente\n"+(3-A)+" intentos restantes");
62         jtausuario.setText(null);
63         pscontrasenia.setText(null);
64     }
65     if (A == 3){
66         dispose();
67         JOptionPane.showMessageDialog(null,"Cuenta Bloqueada!");
68     }
69     }catch (NumberFormatException nfe){
70         System.out.println(" Digita Unicamente numeros en la contraseña "+nfe);
71     }
72 }
```

Nombre de Usuario: Omar

Contraseña (Numeros): ....

CANCELAR ACEPTAR

Contraseña:  
(Letras)

CuentaUser (debug) \* Debugger Console \*

debug:  
Digita Unicamente numeros en la contraseña java.lang.NumberFormatException: For input string: "omar"

## EXCEPCION 1

Se establece calificación de cada unidad, materia y semestre.  
Y se guarda con el botón establecido

```
public void actionPerformed(ActionEvent ae) {
    //U1 = 0; U2 = 0; U3 = 0; U4 = 0; U5 = 0;
    try{
        if (ret == 1){
            U1=0;
            for (int y=0; y<30; y++){
                String O1 = cubito1[y].getText();
                P1 = Byte.parseByte(O1);
                U1+=P1;
            }
            g1= U1;
            //System.out.println(g1);
            if (ret == 2){
                U2=0;
                for (int y=0; y<30; y++){
                    String O2 = cubito2[y].getText();
                    P2 = Byte.parseByte(O2);
                    U2+=P2;
                }
                g2= U2;
                if (ret == 3){
                    U3=0;
                    for (int y=0; y<30; y++){
                        String O3 = cubito3[y].getText();
                        P3 = Byte.parseByte(O3);
                        U3+=P3;
                    }
                    g3= U3;
                    if (ret == 4){
                        U4=0;
                        for (int y=0; y<35; y++){
                            String O4 = cubito4[y].getText();
                            P4 = Byte.parseByte(O4);
                            U4+=P4;
                        }
                        g4= U4;
                        if (ret == 5){
                            U5=0;
                            for (int y=0; y<35; y++){
                                String O5 = cubito5[y].getText();
                                P5 = Byte.parseByte(O5);
                                U5+=P5;
                            }
                            g5= U5;
                        }catch(NumberFormatException nfe){
                            System.out.println("Dijite solo numeros del 1-100 "+nfe);
                        }
                    }
                }
            }
        }
    }
}
```

Aquí sucedió  
un error al  
poner un cero  
y se puso una  
"0"

CuentaUser (debug) x Debugger Console x  
debug:  
Dijite solo numeros del 1-100 java.lang.NumberFormatException: For input string: "90"

## EXCEPCION 2



PROMEDIO SEMESTRES

1er Semestre

calificacion 1er Semestre 90.0

Aprobado

GUARDAR  
CANCELAR  
LIMPIAR  
PROMEDIO

T. Base Datos	99	99	99	99
Simulacion:	99	99	99	99
Fund. Ing. Soft.:	99	99	99	99
Leng. Interfaz:	99	99	99	99
Conta. Finac.:	99	99	99	99

Ahora podemos visualizar el promedio de cada semestre guardado, en el apartado de promedio y conocer si estamos aprobados en el o no.

PROMEDIO SEMESTRES

2do Semestre

calificacion 2do Semestre 70.0

Aprobado

PROMEDIO SEMESTRES

3er Semestre

calificacion 3er Semestre 67.0

Reprobado

PROMEDIO SEMESTRES

4to Semestre

calificacion 4to Semestre 78.0

Aprobado

PROMEDIO SEMESTRES

5to Semestre

calificacion 5to Semestre 69.0

Reprobado

Entrada

¿ Cuantos semestres llevas cursando (1-5)?

0

Acceptar Cancelar

GUARDAR  
CANCELAR  
LIMPIAR  
PROMEDIO

Calculo Dif.:	90	90	90	90
Quimica:	90	90	90	90
Fund. DE Pro.	90	90	90	90
Mat. Discretas:	90	90	90	90
Desa. Sust.:	90	90	90	90
Fund. de Inv.:	90	90	90	90

Sin embargo, antes debemos de introducir la cantidad de materias que registramos del 1 al 5 para saber si eres acreditador a la beca o no, pero nos equivocamos poniendo "0"

CuentaUser (debug) x Debugger Console x

debug:

Por favor indique un numero del 'UNO al CINCO'

```
public void premio() {
    f1=g1/30;
    f2=g2/30;
    f3=g3/30;
    f4=g4/31;
    f5=g5/31;
    try{
        gt = (f1+f2+f3+f4+f5);
        //System.out.println(gt+"\n"+fx);
        gtl= (gt/fx);
        //System.out.println(gtl);
        if (gt > 85){
            JOptionPane.showMessageDialog(null, "Felicidades hasta el momento tienes una calificacion de "+gtl+" \n por lo que conservas tu beca");
            BufferedReader reader = new BufferedReader(new FileReader("C:\\Users\\Oma\\Desktop\\prueba.txt"));
            String firstString = reader.readLine();
            System.out.println(firstString);
        }
        op=1;
    } catch (IOException ex) {
        System.out.println("Error! La ubicacion no se encontro");
        op=1;
    } catch (ArithmeticException ae){
        System.out.println("Por favor indique un numero del 'UNO al CINCO'");
        op=0;
    }
}
```

## EXCEPCION 3

Entrada

Cuantos semestres llevas cursando (1-5)

1

GUARDAR  
CANCELAR  
LIMPIAR  
PROMEDIO

Calculo Dif.: 90 90 90 90 90  
Quimica: 90 90 90 90 90  
Fund. DE Pro. 90 90 90 90 90  
Mat. Discretas: 90 90 90 90 90  
Desa. Sust.: 90 90 90 90 90  
Fund. de Inv.: 90 90 90 90 90

Mensaje

Felicidades hasta el momento tienes una calificación de 90 por lo que conservas tu beca de aprovechamiento

Aceptar

CuentaUser (debug) x Debugger Console x

debug:  
Error! La ubicacion no se encontro

Si digitamos un numero correcto nos debería de mandar un apartado de un documento que este en el computador, sin embargo, la dirección está mal

```
public void premio() {  
    f1=g1/30;  
    f2=g2/30;  
    f3=g3/30;  
    f4=g4/31;  
    f5=g5/31;  
    try{  
        gt = (f1+f2+f3+f4+f5);  
        //System.out.println(gt+"\n"+fx);  
        gtl= (gt/fx);  
        //System.out.println(gtl);  
        if (gt > 85){  
            JOptionPane.showMessageDialog(null, "Felicidades hasta el momento tienes una calificacion de "+gtl+" \n por lo que conservas tu beca  
            BufferedReader reader = new BufferedReader(new FileReader("C:\\Users\\Oma\\Desktop\\prueba.txt"));  
            String firstString = reader.readLine();  
            System.out.println(firstString);  
        }  
        op=1;  
    } catch (IOException ex) {  
        System.out.println("Error! La ubicacion no se encontro");  
        op=1;  
    } catch (ArithmeticException ae){  
        System.out.println("Por favor indique un numero del 'UNO al CINCO'");  
        op=0;  
    }  
}
```

## EXCEPCION 4





REGISTRO DE CALIFICACIONES

CARRERA: Ing. Sistemas

SEMESTRE: 5to Semestre

GUARDAR  
CANCELAR  
LIMPIAR  
PROMEDIO

Leng. Automatas.: ☐ ☐ ☐ ☐

Redes Comp.: ☐ ☐ ☐ ☐

T. Base Datos: ☐ ☐ ☐ ☐

Simulacion: ☐ ☐ ☐ ☐

Fund. Ing. Soft.: ☐ ☐ ☐ ☐

Leng. Interfaz: ☐ ☐ ☐ ☐

Conta. Finac.: ☐ ☐ ☐ ☐

Con el botón limpiar, borramos todas las calificaciones anteriormente anotadas, y al presionar nuevamente el botón promedio, nos damos cuenta que igual se eliminan.

PROMEDIO SEMESTRES

1er Semestre

calificacion 1er Semestre 0.0

Establece calificacion

GUARDAR  
CANCELAR  
LIMPIAR  
PROMEDIO

Fund. De Tel. ☐ ☐ ☐ ☐

Inv. de Op.: ☐ ☐ ☐ ☐

S. Operativos: ☐ ☐ ☐ ☐

Prin. Elec. Y Ap. Dig: ☐ ☐ ☐ ☐

REGISTRO DE CALIFICACIONES

CARRERA: Ing. Sistemas

SEMESTRE: 2do Semestre

GUARDAR  
CANCELAR  
LIMPIAR  
PROMEDIO

Calculo Int.: ☐ ☐ ☐ ☐

Fisica: ☐ ☐ ☐ ☐

P.O.O: ☐ ☐ ☐ ☐

Prob. Y Est.: ☐ ☐ ☐ ☐

T. Admin.: ☐ ☐ ☐ ☐

Algebra L.: ☐ ☐ ☐ ☐

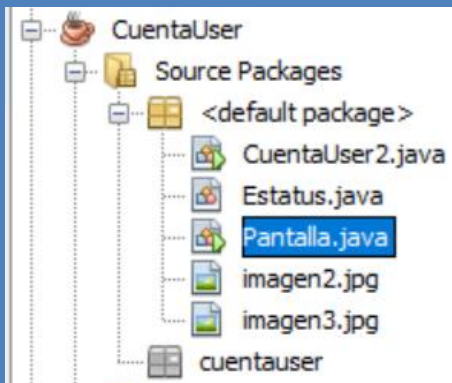
Al cerrar nos manda al apartado para poner nuevamente la contraseña.

Iniciar Sesión

Nombre de Usuario:

Contraseña:

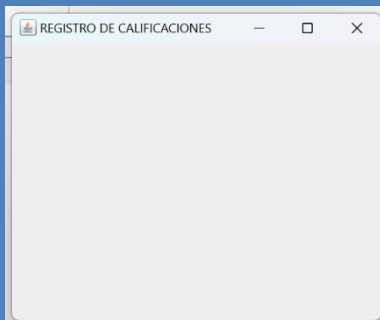
CANCELAR ACEPTAR



```
public class ImageFondo extends JPanel{

    private Image imagen;

    public void paint(Graphics g){
        try{
            imagen = new ImageIcon(getClass().getResource("imagen1.jpg")).getImage();
            g.drawImage(imagen, 0, 0, getWidth(), getHeight(), this);
            setOpaque(false);
            super.paint(g);
        }catch (NullPointerException npe){
            System.out.printf("No esta la imagen \n"+npe+"\n");
        }
    }
}
```



CuentaUser (debug) x Debugger Console x

```
debug:
No esta la imagen
java.lang.NullPointerException
```

Pensamos que introducimos la imagen numero 1, por lo que no la tenemos. Y no nos muestra nada el frame

## EXCEPCION 5

## V. Conclusiones: (MEDIA CUARTILLA)

Al momento de estar programando nos encontramos con muchos errores, ya sean lógicos, de sintaxis entre otros, pues siempre estarán presentes, pero por que no entender su causa, y así darles una solución más excepcional.

Los usuarios suelen cometer la mayoría de estos errores que el programa no suele entender, pues si este pidió peras, por que das manzanas, y es que a veces tenemos errores de dedo, pero, ¿cuando no depende del dedo? La unidad “Excepciones” es muy curiosa, por que tratamos sobre lo que todos tenemos... así es, errores, malignos errores, y gracias a las excepciones podemos evitar que el programa se detenga, ya que es muy visto que tienes que hacer todo de nuevo por una equivocación y claro, es muy muy frustrante.

Tenemos diferentes excepciones, como lo son aritméticos, de ubicación, de carácter vacío, entre muchas, pues estas nos ayudan a seguir corriendo el programa, y saber en que nos equivocamos para corregirlo.

Podemos decir, que las excepciones son muy importantes para la creación de programas.