UNLaM



Escuela de Formación Contínua

**PROGRAMACIÓN AVANZADA I**

**1er Parcial**

Apellido y Nombre **07 / 06 / 2021**

DNI: **Calificación :**

1. La clase ***MiCalendario*** vista en clase:
   1. Se utiliza para adaptar la clase ***GregorianCalendar*** a nuestras necesidades.
   2. Es una clase que admite que se creen objetos con valores de fecha inválidos.
   3. Permite la creación de objetos que también son de tipo ***GregorianCalendar***.
   4. a, b y c son correctas.
   5. a y b son correctas.
   6. a y c son correctas. Esta es correcta
   7. Ninguna es correcta.

# ==================================================================================

1. Complete el siguiente código (tomado del ***ConversorGenerico*** visto en clase), de modo que si el número contenido en la variable ***valor1TextField***, tuviera coma en lugar de punto para separar la parte entera de la decimal, no impida al método la correcta conversión:

private void convertirAValor2(){

String valor1String = valor1TextField.getText(); Double valor1Double;

try {

valor1Double = Double.valueOf(valor1String);

} catch (NumberFormatException e) {

JOptionPane.showMessageDialog(this, “No se pudo convetir”, "ERROR", JOptionPane.ERROR\_MESSAGE);

valor1TextField.requestFocus(); return;

}

Double valor2Double = conversorSelecciondo.convertirValor1Valor2(valor1Double); valor2TextField.setText(String.format("%.2f", valor2Double));

}

Respuesta

private void convertirAValor2() {

String valor1String = valor1TextField.getText();

Double valor1Double;

// Reemplazar coma por punto si existe en el valor1String

valor1String = valor1String.replace(",", ".");

try {

valor1Double = Double.valueOf(valor1String);

} catch (NumberFormatException e) {

JOptionPane.showMessageDialog(this, "No se pudo convertir", "ERROR", JOptionPane.ERROR\_MESSAGE);

valor1TextField.requestFocus();

return;

}

Double valor2Double = conversorSelecciondo.convertirValor1Valor2(valor1Double);

valor2TextField.setText(String.format("%.2f", valor2Double));

}

# ==================================================================================

1. Una **excepción** es:
   1. Un código que retorna un método cuando detecta un error.
   2. Un método que crea un objeto que se lanza al detectarse una condición errónea.
   3. Un bloque ***try/catch***.
   4. Un objeto que es devuelto, por medio de la sentencia ***throw***, guardándose en él información del error ocurrido.
   5. a, b, c y d son correctas. Esta es correcta
   6. a, b, c y d son incorrectas.

# ==================================================================================

1. Desarrolle un constructor de la clase ***Alumno*** que reciba todos los parámetros necesarios para inicializar sus atributos y los de sus clase padre ***Persona***. donde ***Persona*** posee los siguientes atributos:

private int dni; private String apyn;

private MiCalendario fechaNac;

Y que ***Alumno*** posee los siguientes atributos: private double promedio; private int cantMatAprob;

Escriba al menos 2(dos) validaciones.

public class Alumno extends Persona {

private double promedio;

private int cantMatAprob;

public Alumno(int dni, String apyn, MiCalendario fechaNac, double promedio, int cantMatAprob) {

super(dni, apyn, fechaNac);

if (cantMatAprob >= 0) {

this.cantMatAprob = cantMatAprob;

} else {

throw new IllegalArgumentException("La cantidad de materias aprobadas no puede ser negativa.");

}

if (promedio >= 0 && promedio <= 10) {

this.promedio = promedio;

} else {

throw new IllegalArgumentException("El promedio debe estar entre 0 y 10.");

}

}

}

# ==================================================================================

1. Dado el siguiente código:

public abstract class Animal

{

public abstract String hablar();

}

Desarrolle las clases ***Perro*** y ***Gato***, las cuales hereden de ***Animal***.

public class Perro extends Animal {

@Override

public String hablar() {

return "Woof!";

}

}

public class Gato extends Animal {

@Override

public String hablar() {

return "Meow!";

}

}

# ==================================================================================