

Integrador Java + MySQL

Total de puntos 97/100 ?

Hemos llegado a la mitad del curso, **es el momento de validar lo que has aprendido hasta ahora**. ¡Aprovecha este espacio para chequear qué tal vienes hasta aquí!

Antes

de resolver este cuestionario tenes que haber realizado el ejercicio de los candados de MySQL. Por otro lado recomendamos revisar previas guías y videos si es necesario para realizar el cuestionario con los conceptos refrescados. Si ya lo hiciste, ¡adelante!

Tendrás un único intento

¡Buena suerte!

Correo electrónico *

jimematthies@gmail.com

0 de 0 puntos

Correo Electrónico *

El mismo correo con el que Ingresas a tu Aula Virtual (con el que estás registrado en Egg)

jimematthies@gmail.com



DNI *Escribe tu cédula de identidad **sin puntos, ni espacios**

166247985

Nombre y Apellido *

Maria Jimena Matthies Veliz

Java Introducción

11 de 11 puntos

✓ Tipos de datos en Java: *

2/2

- ☐ int, double, boolean, String, void, null
- ☐ Cadena, entero, real, lógico
- ☒ byte, short, int, long, float, double, boolean, char, String



✓ **Qué hace el siguiente código? ***

2/2



```
1 int aux = 0;
2 aux++;
```

- ☐ Multiplica por 2 a la variable aux
- ☒ Incrementa en 1 a la variable aux
- ☐ Agrega una nueva variable aux



✓ **Las variables siempre deben especificar de qué tipo son al ser *1/1 declaradas.**

- ☒ Verdadero
- ☐ Falso



✓ **Cuál es la salida del siguiente código? ***

2/2



```
1  int num1 = 6;  
2  int num2 = 8;  
3  
4  System.out.println(num1==num2);
```

- ☐ True
- ☒ False
- ☐ 6
- ☐ 8



✓ **Cual de los siguientes métodos NO es parte de la clase String?**

***2/2**

- ☒ .stringLength()
- ☐ .isEmpty()
- ☐ .toUpperCase()
- ☐ concat()



✓ **El método main puede estar o no en un proyecto Java funcional.**

*2/2

☐ Verdadero

☒ Falso



POO

21 de 24 puntos

✓ **¿De que se encarga Clase servicio o control? ***

2/2

☐ Se encarga de controlar que el programa no consuma memoria al instanciar objetos.

☐ Se encarga de recibir los datos que ingresa el usuario por teclado

☒ Se encarga de crear y modificar objetos, teniendo cada clase de entidad una clase control. ✓

✓ **Cuál es la diferencia entre clases y objetos? ***

2/2

☒ Las clases son “moldes” y los objetos son modelados a partir de esos moldes, también llamados “instancias” ✓

☐ Los objetos son abstracciones del mundo real, las clases son de lunes a jueves

☐ Las clases y los objetos son los mismo



✓ **Cuáles son los modificadores de acceso? ***

2/2

- ☐ Public, notPublic
- ☐ Public, static, abstract
- ☒ Public, private, protected



✓ **Qué significa POO? ***

1/1

- ☐ Programación, Observación y Orientación
- ☒ Programación Orientada a Objetos
- ☐ Programación Objetiva Orientada



✓ **Qué es el encapsulamiento? ***

2/2

- ☐ Hace referencia a los métodos constructores con y sin parámetros
- ☐ Hace referencia a separar el código en “cápsulas” para mejor lectura del mismo.
- ☒ Hace referencia a crear métodos para obtener y modificar información de un objeto, manteniendo oculto lo que hace un objeto por dentro.



✓ **Qué es un constructor? ***

2/2

- ☒ Es un método que nos permite instanciar un objeto de la clase donde se encuentra ✓
- ☐ Un método que nos define qué atributos tiene la clase
- ☐ Un método void

✓ **Qué son los atributos? ***

2/2

- ☒ Variables que guardan información sobre el objeto ✓
- ☐ Funciones que el objeto puede realizar
- ☐ Métodos de consulta o modificación de un objeto



✗ **Cuál es la salida de este código? ***

0/3



```
1 Persona p1 = new Persona();
2 Persona p2 = new Persona();
3
4 p1.setNombre("Albus Dumbledore");
5 p2.setNombre(p1.getNombre());
6
7 System.out.println(p1.equals(p2));
8 System.out.println(p1.getNombre().equals(p2.getNombre()));
```

- ☐ False, False
- ☐ True, False
- ☒ True, True
- ☐ False, True



Respuesta correcta

- ☒ False, True



✓ **Cuál de estos servicios trabaja de la manera correcta? ***

3/3



```
1 public void conocerNombre(Persona x){  
2     System.out.println("La persona se llama " + x.getNombre());  
3 }  
4  
5 public void conocerNombre(String nombre){  
6     System.out.println("La persona se llama " + nombre);  
7 }
```

- ☒ El que obtiene el nombre a través de un objeto Persona
- ☐ El que obtiene el nombre desde un String



✓ **Qué hace el siguiente código? ***

3/3



```
1 Persona p1 = new Persona();
```

- ☐ Crea una nueva clase llamada Persona
- ☐ Duplica un objeto persona
- ☒ Instancia un objeto de tipo Persona llamado p1



✓ **Cómo accedo a los atributos de un objeto? ***

2/2

- ☐ Con el símbolo ! luego del nombre del objeto
- ☐ Con el símbolo _ luego del nombre del objeto
- ☒ Con el símbolo . luego del nombre del objeto



Colecciones

17 de 17 puntos

✓ **Las listas son colecciones de objetos ordenados por posición donde los elementos pueden repetirse. ***2/2

- ☐ Falso
- ☒ Verdadero



✓ **Un framework es un marco de trabajo el cual contiene un conjunto estandarizado de conceptos, prácticas y criterios para hacer frente a un tipo de problemática particular y resolver nuevos problemas de índole similar. ***2/2

- ☒ Verdadero
- ☐ Falso



✓ **Cómo se define el tamaño de una colección? ***

3/3

- ☐ Con el método size()
- ☒ Las colecciones son dinámicas, su tamaño depende de la cantidad de objetos que le agreguemos o quitemos ✓
- ☐ Se define al declarar la colección

✓ **Cuál de las siguientes herramientas nos facilita recorrer una lista? *2/2**

- ☒ Bucle ForEach ✓
- ☐ Un Switch con condicionales en cada caso
- ☐ Bucle For

✓ **Qué puede almacenar una colección? ***

2/2

- ☒ Objetos de la misma clase ✓
- ☐ Datos primitivos, como char e int
- ☐ Objetos mezclados sin importar de qué clase son



✓ **Los conjuntos o Sets pueden contener elementos repetidos, simplemente se duplica el valor** *2/2

☒ Falso



☐ Verdadero

✓ **Los mapas son colecciones de Llave/Valor. Las llaves son únicas pero los valores sí pueden repetirse.** *2/2

☐ Falso

☒ Verdadero



✓ **El Java Collection Framework es: *** 2/2

☐ Una base de datos

☐ Un conjunto de objetos

☒ Un conjunto de clases e interfaz



Relaciones entre Clases

12 de 12 puntos

✓ **En una relación de agregación, un objeto depende del otro, no pueden existir individualmente.** *2/2

☐ Verdadero

☒ Falso



✓ **Las clases no actúan aisladas entre sí, al contrario, las clases están relacionadas unas con otras.** *2/2

☒ Verdadero



☐ Falso

✓ **Las relaciones entre clases realmente significan que una clase contiene una referencia a un objeto u objetos, de la otra clase en la forma de un atributo.** *2/2

☒ Verdadero



☐ Falso

✓ **Las relaciones son siempre unidireccionales *** 2/2

☐ Verdadero

☒ Falso



✓ **La composición es un tipo de relación donde un objeto complejo es conformado por objetos más pequeños.** *2/2

☐ Falso

☒ Verdadero



✓ Las relaciones pueden ser *

2/2

- ☒ Uno a Uno, Cero a Uno, Uno a Muchos y Cero a Muchos
- ☐ Uno a Uno solamente
- ☐ Uno a Uno y Muchos a Muchos



Herencia

15 de 15 puntos

✓ En el siguiente código ¿cuál es la clase padre? *

3/3

```
1 public class Persona extends SerVivo{
2
3     String nombre;
4
5 }
```

- ☐ Java.util
- ☐ Persona
- ☒ SerVivo
- ☐ public class



✓ Una clase hija puede heredar de muchas clases padres. *

1/1

- ☐ Verdadero
- ☒ Falso



✓ **Una clase padre puede heredar a muchas clases hijas.** * 2/2

☒ Verdadero



☐ Falso

✓ **La Sub Clase hereda atributos y métodos.** * 1/1

☒ Verdadero



☐ Falso

✓ **Para qué sirve el modificador Protected?** * 2/2

☒ Para que los atributos sean accesibles sólo por las clases que heredan sin usar getters ni setters



☐ Para proteger el código para que no sea borrado

☐ Para que suene a que sabemos más que los demás

✓ **Cuándo usamos la palabra reservada super?** * 2/2

☐ Cuando queremos acceder a los métodos, atributos y constructores de la clase inferior

☒ Cuando queremos acceder a los métodos, atributos y constructores de la clase superior



☐ Cuando queremos referirnos a una clase que está muy bien hecha



✓ **Qué hace la anotación @Override? ***

1/1

- ☐ Desbloquea nuevos métodos disponibles en Java
- ☒ Permite que la clase hija haga funcionar un método de manera diferente a la clase padre ✓
- ☐ Permite que la clase hija elimine métodos que considere innecesarios

✓ **Qué es la Super Clase? ***

2/2

- ☒ Como se llama a la clase superior(Padre o Madre) ✓
- ☐ Una clase con superpoderes
- ☐ Una clase con muchos métodos

✓ **La Sub Clase hereda el constructor ***

1/1

- ☒ Falso ✓
- ☐ Verdadero

Manejo de Excepciones

13 de 13 puntos



✓ Cuando se lanza una excepción no hay nada más que hacer, termina el programa. *1/1

☐ Verdadero

☒ Falso



✓Cuál es el resultado del siguiente código? siguiendo el ejemplo que venimos viendo *4/4

```
1 Persona p1 = new Persona();
2 String nombre = null;
3
4 try {
5     if (!nombre.equals(null)) {
6         p1.setNombre(nombre);
7     }
8 } catch (Exception e) {
9     System.out.println("El nombre no puede ser nulo");
10 }
```

☒ Se imprime por consola "El nombre no puede ser nulo"

☐ El nombre de p1 es null

☐ El nombre de p1 queda vacío



✓ Un objeto Exception contiene información sobre un error ocurrido. *2/2

☐ Falso

☒ Verdadero



✓ Si una excepción se produce en el bloque catch, puede atraparse en el bloque try y manejarla.

*2/2

☐ Verdadero

☒ Falso



✓ Para aclarar que un método puede lanzar un error se coloca la palabra “extends”.

*1/1

☐ Verdadero

☒ Falso



✓ Una excepción es un evento que ocurre durante la ejecución de un programa que interrumpe el flujo normal de las instrucciones del programa.

*1/1

☒ Verdadero

☐ Falso



✓ Sólo se puede usar 1 bloque catch por bloque try.*

2/2

☐ Verdadero

☒ Falso



MySQL

8 de 8 puntos



Resultados del ejercicio de los candados

✓ **Candado B ***

1/1

- ☐ Clave: 4830
- ☐ Clave: 4380
- ☐ Clave: 3830
- ☒ Clave: 3480

✓ **Candado C ***

1/1

- ☒ Posición 1
- ☐ Posición 2
- ☐ Posición 3
- ☐ Posición 4

✓ **Candado C ***

1/1

- ☒ Clave: 631
- ☐ Clave: 963
- ☐ Clave: 145
- ☐ Clave: 601



✓ **Candado A ***

1/1

- ☒ Clave: 14043
- ☐ Clave: 16832
- ☐ Clave: 15935
- ☐ Clave: 13539



✓ **Candado B ***

1/1

- ☐ Posición 1
- ☐ Posición 2
- ☒ Posición 3
- ☐ Posición 4



✓ **Candado D ***

1/1

- ☒ Clave: 191
- ☐ Clave: 1063
- ☐ Clave: 153
- ☐ Clave: 101



✓ **Candado D ***

1/1

- ☐ Posición 1
- ☐ Posición 2
- ☐ Posición 3
- ☒ Posición 4



✓ **Candado A ***

1/1

- ☐ Posición 1
- ☒ Posición 2
- ☐ Posición 3
- ☐ Posición 4



El formulario se creó en Egg Cooperation.

Google Formularios





