



# CURSO DE PROGRAMACIÓN



## Integrador Java + MySQL + JDBC + JPA

Puntos totales **100/100** ?

### Correo \*

spetrini@gmail.com

### Datos personales

0 de 0 puntos

### DNI \*

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

26657356

### Correo Electrónico \*

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

spetrini@gmail.com

### Nombre y Apellido \*

Sebastián A. Petrini



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

☐ Verdadero

☒ Falso



✓ **Cómo se define el tamaño de una colección? \*** 3/3

☒ Las colecciones son dinámicas, su tamaño depende de la cantidad de objetos que le agreguemos o quitemos ✓

☐ Se define al declarar la colección

☐ Con el método size()

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

☒ Bucle ForEach ✓

☐ Bucle For

☐ Un Switch con condicionales en cada caso

✓ **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

☒ Un conjunto de clases e interfaces



☐ Una base de datos

☐ Un conjunto de objetos

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

☒ Verdadero



☐ Falso

✓ **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**

☐ Falso

☒ Verdadero



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

2/2

☐ Datos primitivos, como char e int

☐ Objetos mezclados sin importar de qué clase son

☒ Objetos de la misma clase



✓ **Las relaciones son siempre unidireccionales \***

2/2

☒ Falso



☐ Verdadero

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

☐ Falso

☒ Verdadero



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

☐ Falso

☒ Verdadero



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

☒ Verdadero



☐ Falso

✓ **Las relaciones pueden ser <sup>\*</sup>** 2/2

☐ Uno a Uno y Muchos a Muchos

☐ Uno a Uno solamente

☒ Uno a Uno, Uno a Muchos, Muchos a Muchos, Muchos a Uno



## Herencia

15 de 15 puntos

✓ **Cuándo usamos la palabra reservada super? <sup>\*</sup>** 2/2

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

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



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



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

1/1

- ☒ 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
- ☐ Desbloquea nuevos métodos disponibles en Java

✓ **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

✓ **Qué es la Super Clase? \***

2/2

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



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

1/1

☐ 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



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

3/3

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

☐ Persona

☐ public class

☒ SerVivo



☐ Java.util

## Manejo de Excepciones

13 de 13 puntos

✓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 }
```

☐ El nombre de p1 queda vacío

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



☐ El nombre de p1 es null





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

2/2

☐ 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

☐ Falso

☒ Verdadero



✓ Para aclarar que un método puede lanzar un error se coloca la palabra "extends". \*1/1

☒ Falso



☐ Verdadero

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

☒ Verdadero



☐ Falso



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

\*2/2

☐ Verdadero

☒ Falso



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

\*1/1

☐ Verdadero

☒ Falso



## MySQL

8 de 8 puntos

Resultados del ejercicio de los candados

✓ Candado B \*

1/1

☐ Clave: 4380

☒ Clave: 3480



☐ Clave: 3830

☐ Clave: 4830



✓ **Candado C \***

1/1

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

✓ **Candado B \***

1/1

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

✓ **Candado A \***

1/1

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



✓ **Candado D \***

1/1

☐ Posición 1

☐ Posición 2

☐ Posición 3

☒ Posición 4



✓ **Candado C \***

1/1

☒ Posición 1

☐ Posición 2

☐ Posición 3

☐ Posición 4



✓ **Candado D \***

1/1

☒ Clave: 191

☐ Clave: 1063

☐ Clave: 153

☐ Clave: 101



✓ **Candado A \***

1/1

☐ Clave: 15935

☒ Clave: 14043 ✓

☐ Clave: 13539

☐ Clave: 16832

## JDBC

20 de 20 puntos

✓ ¿Cuál de las siguientes afirmaciones es cierta acerca de las transacciones en JDBC? \*2/2

☐ A) Las transacciones solo se utilizan para consultas SELECT.

☐ B) Las transacciones solo se utilizan para consultas de modificación de datos (INSERT, UPDATE, DELETE).

☒ C) Las transacciones permiten agrupar múltiples consultas en una única operación atómica. ✓

☐ D) Las transacciones solo son necesarias cuando se utilizan controladores de base de datos específicos.

✓ ¿Cuál de las siguientes declaraciones es cierta acerca de los controladores (drivers) JDBC? \*2/2

☐ C) Los controladores JDBC son necesarios solo para la conexión inicial a la base de datos.

☐ A) Los controladores JDBC son responsables de la administración de transacciones.

☐ B) Los controladores JDBC son proporcionados por el servidor de la base de datos.

☒ D) Los controladores JDBC son interfaces utilizadas por la aplicación para interactuar con una base de datos. ✓



✓ ¿Cuál es la forma más segura de ejecutar consultas SQL en JDBC \*2/2  
para prevenir ataques de inyección de SQL?

- ☐ D) Utilizar el objeto ResultSet para obtener resultados de consultas.
- ☐ B) Utilizar el objeto Statement para ejecutar consultas sin parámetros.
- ☐ A) Concatenar los parámetros de consulta directamente en la cadena SQL.
- ☒ C) Utilizar el objeto PreparedStatement con parámetros vinculados. ✓

✓ ¿Cuál es el propósito principal de JDBC? \* 2/2

- ☒ B) Establecer la conexión con una base de datos. ✓
- ☐ C) Mapear objetos a tablas de base de datos.
- ☐ D) Realizar operaciones CRUD en una base de datos.
- ☐ A) Crear consultas SQL.

✓ ¿Cuál es el propósito del objeto ResultSet en JDBC? 2/2

- ☐ C) Ejecuta una consulta SQL.
- ☐ D) Define los parámetros de una consulta parametrizada.
- ☒ B) Almacena el resultado de una consulta SQL. ✓
- ☐ A) Representa una conexión a una base de datos.



✓ ¿Cuál de las siguientes interfaces de JDBC se utiliza para ejecutar \*2/2 consultas parametrizadas?

☐ C) Connection.

☒ D) PreparedStatement. ✓

☐ B) ResultSet.

☐ A) Statement.

✓ ¿Cuál es la forma correcta de cerrar una conexión JDBC \*2/2 correctamente?

☐ B) Llamar al método close() en el objeto ResultSet.

☒ C) Llamar al método close() en el objeto Connection. ✓

☐ D) Llamar al método close() en todos los objetos anteriores.

☐ A) Llamar al método close() en el objeto Statement.

✓ ¿Cuál es el propósito del objeto Connection en JDBC? \* 2/2

☐ D) Proporcionar métodos para obtener metadatos de la base de datos.

☐ B) Almacenar los resultados de una consulta.

☒ C) Representar una conexión física a una base de datos. ✓

☐ A) Ejecutar consultas SQL.



✓ ¿Cuál es la forma más eficiente de realizar múltiples inserciones en una base de datos utilizando JDBC? \*2/2

- ☐ B) Utilizar un bucle y ejecutar consultas INSERT individuales.
- ☐ A) Ejecutar una consulta INSERT para cada fila a insertar.
- ☐ D) Utilizar el objeto BatchStatement para ejecutar consultas en lote.
- ☒ C) Utilizar un objeto PreparedStatement y una transacción. ✓

✓ ¿Cuál es la forma correcta de manejar excepciones en JDBC? \* 2/2

- ☒ C) Capturar excepciones específicas de JDBC y manejarlas adecuadamente. ✓
- ☐ A) Ignorar las excepciones y continuar con la ejecución del programa.
- ☐ D) Utilizar el bloque finally para liberar recursos, sin importar si se produce una excepción o no.
- ☐ B) Lanzar una nueva excepción personalizada en cada método de JDBC.

## JPA

15 de 15 puntos

✓ ¿Cuál de las siguientes afirmaciones describe mejor JPA (Java Persistence API)? \*3/3

- ☐ D) Es un framework para el desarrollo de interfaces de usuario en Java.
- ☐ B) Es una biblioteca para el acceso a bases de datos NoSQL.
- ☒ C) Es una especificación de Java para el mapeo objeto-relacional. ✓
- ☐ A) Es un lenguaje de consulta para bases de datos relacionales.





✓ ¿Cuál de las siguientes anotaciones se utiliza para marcar una propiedad como una clave primaria en JPA? \*2/2

- ☐ C) @GeneratedValue
- ☐ D) @Column
- ☐ B) @PrimaryKey
- ☒ A) @Id



✓ ¿Cómo se llama la interface medular de JPA? \* 3/3

- ☒ Entity Manager
- ☐ Manager de Entidad
- ☐ ORM
- ☐ Manager Factory



✓ ¿Cuál de las siguientes anotaciones se utiliza para marcar una clase como una entidad en JPA? \*2/2

- ☐ C) @PersistenceContext
- ☒ A) @Entity
- ☐ B) @Table
- ☐ D) @OneToMany



✓ ¿Cuál de las siguientes anotaciones se utiliza para establecer una **\*2/2** relación de uno a muchos en JPA?

☐ C) @OneToOne

☐ D) @ManyToMany

☒ A) @OneToMany



☐ B) @ManyToOne

✓ ¿Cuál s el método que se encarga de buscar y devolver una **\*3/3** Entidad?

☐ .persist()

☐ .getTransaction()

☒ .find()



☐ .merge()

Este formulario se creó en Egg Cooperation.

Google Formularios

