



# CURSO DE PROGRAMACIÓN



## Integrador Java + MySQL + JDBC + JPA

Total de puntos **87/100** ?

### Correo electrónico \*

claudia.alvarez402@gmail.com

### Datos personales

0 de 0 puntos

### DNI \*

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

35635995

### Nombre y Apellido \*

Claudia Alvarez Ginestet



### Correo Electrónico \*

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

claudia.alvarez402@gmail.com

### Colecciones

15 de 17 puntos

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



✓ **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()

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

- ☐ Verdadero
- ☒ Falso ✓

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

0/2

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

Respuesta correcta

- ☒ Objetos de la misma clase



✓ **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



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

\*2/2

☐ Un Switch con condicionales en cada caso

☒ Bucle ForEach



☐ Bucle For

## Relaciones entre Clases

12 de 12 puntos

✓ **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



✓ **Las relaciones pueden ser \***

2/2

☐ Uno a Uno y Muchos a Muchos

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



☐ Uno a Uno solamente

✓ **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



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

\*2/2

☒ Verdadero



☐ Falso

## Herencia

14 de 15 puntos

✓ **Cuándo usamos la palabra reservada super? \***

2/2

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

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

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



✓ **Qué es la Super Clase? \***

2/2

☒ Como se llama a la clase superior(Padre o Madre)



☐ Una clase con muchos métodos

☐ Una clase con superpoderes



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

1/1

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

✓ **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 }
```

☐ public class

☐ Persona

☒ SerVivo



☐ Java.util

✓ La Sub Clase hereda atributos y métodos. \*

1/1

☒ Verdadero



☐ Falso

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

1/1

☐ Verdadero

☒ Falso





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

0/1

☐ Falso

☒ Verdadero



Respuesta correcta

☒ Falso

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

2/2

☒ Verdadero



☐ Falso

## Manejo de Excepciones

13 de 13 puntos

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

\*2/2

☐ Falso

☒ Verdadero



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

☒ Falso



☐ Verdadero

✓ **Si una excepción se produce en el bloque catch, puede atraparse en el bloque try y manejarla.** \*2/2

☐ Verdadero

☒ Falso



✓ **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

✓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 es null

☐ El nombre de p1 queda vacío

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



MySQL

8 de 8 puntos

Resultados del ejercicio de los candados



✓ **Candado B \***

1/1

☐ Clave: 4380

☒ Clave: 3480



☐ Clave: 4830

☐ Clave: 3830

✓ **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 A \***

1/1

☐ Clave: 15935

☐ Clave: 16832

☐ Clave: 13539

☒ Clave: 14043



✓ **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 C \***

1/1

☒ Clave: 631



☐ Clave: 963

☐ Clave: 145

☐ Clave: 601

✓ **Candado D \***

1/1

☒ Clave: 191



☐ Clave: 1063

☐ Clave: 153

☐ Clave: 101

## JDBC

14 de 20 puntos

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

☐ A) Concatenar los parámetros de consulta directamente en la cadena SQL.

☒ C) Utilizar el objeto PreparedStatement con parámetros vinculados.



☐ D) Utilizar el objeto ResultSet para obtener resultados de consultas.

☐ B) Utilizar el objeto Statement para ejecutar consultas sin parámetros.



✗ ¿Cuál de las siguientes declaraciones es cierta acerca de los controladores (drivers) JDBC? \*0/2

- ☐ A) Los controladores JDBC son responsables de la administración de transacciones.
- ☐ D) Los controladores JDBC son interfaces utilizadas por la aplicación para interactuar con una base de datos.
- ☒ B) Los controladores JDBC son proporcionados por el servidor de la base de datos. ✗
- ☐ C) Los controladores JDBC son necesarios solo para la conexión inicial a la base de datos.

Respuesta correcta

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

✗ ¿Cuál es la forma correcta de cerrar una conexión JDBC correctamente? \*0/2

- ☐ A) Llamar al método close() en el objeto Statement.
- ☐ C) Llamar al método close() en el objeto Connection.
- ☐ B) Llamar al método close() en el objeto ResultSet.
- ☒ D) Llamar al método close() en todos los objetos anteriores. ✗

Respuesta correcta

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



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

2/2

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

✗ ¿Cuál es la forma más eficiente de realizar múltiples inserciones en una base de datos utilizando JDBC? \*0/2

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

Respuesta correcta

- ☒ C) Utilizar un objeto PreparedStatement y una transacción.

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

2/2

- ☐ B) Almacenar los resultados de una consulta.
- ☐ A) Ejecutar consultas SQL.
- ☒ C) Representar una conexión física a una base de datos. ✓
- ☐ D) Proporcionar métodos para obtener metadatos de la base de datos.





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

2/2

☒ B) Establecer la conexión con una base de datos. ✓

☐ D) Realizar operaciones CRUD en una base de datos.

☐ A) Crear consultas SQL.

☐ C) Mapear objetos a tablas de base de datos.

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

2/2

☐ A) Ignorar las excepciones y continuar con la ejecución del programa.

☒ C) Capturar excepciones específicas de JDBC y manejarlas adecuadamente. ✓

☐ B) Lanzar una nueva excepción personalizada en cada método de JDBC.

☐ D) Utilizar el bloque finally para liberar recursos, sin importar si se produce una excepción o no.

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

☐ C) Connection.

☐ B) ResultSet.

☐ A) Statement.

☒ D) PreparedStatement. ✓



✓ ¿Cuál de las siguientes afirmaciones es cierta acerca de las transacciones en JDBC?

\*2/2

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

**JPA**

11 de 15 puntos

✗ ¿Cuál de las siguientes opciones describe mejor el concepto de "carga diferida" (lazy loading) en JPA? \*0/2

- ☐ D) Cargar todas las propiedades de una entidad de forma anticipada.
- ☒ C) Cargar solo las propiedades necesarias de una entidad cuando se accede a ella. ✗
- ☐ A) Cargar solo las entidades relacionadas necesarias cuando se accede a ellas.
- ☐ B) Cargar todas las entidades relacionadas de forma anticipada.

Respuesta correcta

- ☒ A) Cargar solo las entidades relacionadas necesarias cuando se accede a ellas.



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

☐ B) @ManyToOne

☒ A) @OneToMany ✓

☐ C) @OneToOne

☐ D) @ManyToMany

✗ ¿Cuál de las siguientes anotaciones se utiliza para marcar una \*0/2 propiedad como una clave primaria en JPA?

☒ B) @PrimaryKey ✗

☐ D) @Column

☐ C) @GeneratedValue

☐ A) @Id

Respuesta correcta

☒ A) @Id



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

☐ C) @PersistenceContext

☐ B) @Table

☐ D) @OneToMany

☒ A) @Entity ✓

✓ ¿Cuál de las siguientes opciones describe mejor la diferencia entre CascadeType.PERSIST y CascadeType.MERGE en JPA? \*2/2

☐ D) CascadeType.PERSIST guarda solo las entidades modificadas, mientras que CascadeType.MERGE guarda todas las entidades relacionadas.

☐ C) CascadeType.PERSIST realiza una operación de actualización, mientras que CascadeType.MERGE realiza una operación de inserción.

☒ A) CascadeType.PERSIST realiza una operación de inserción, mientras que CascadeType.MERGE realiza una operación de actualización. ✓

☐ B) CascadeType.PERSIST guarda todas las entidades relacionadas, mientras que CascadeType.MERGE guarda solo las entidades modificadas.



✓ ¿Cuál de las siguientes opciones describe mejor el concepto de "cascada" (cascade) en JPA? \*2/2

☒ B) La forma en que se propagan las operaciones en cascada a entidades relacionadas. ✓

☐ A) La forma de almacenar entidades en cascada utilizando JPA.

☐ C) La forma de generar automáticamente identificadores únicos para las entidades.

☐ D) La forma de cargar entidades relacionadas en cascada utilizando JPA.

✓ ¿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.

☒ C) Es una especificación de Java para el mapeo objeto-relacional. ✓

☐ A) Es un lenguaje de consulta para bases de datos relacionales.

☐ B) Es una biblioteca para el acceso a bases de datos NoSQL.

El formulario se creó en Egg Cooperation.

Google Formularios

















