



CURSO DE **PROGRAMACIÓN**



Integrador Java + MySQL + JDBC + JPA

Puntos totales	100/100	?
runios idiales	100/100	E

Canaa *	
Correo *	
spetrini@gmail.com	
Datos personales	0 de 0 puntos
Datos personales	o de o puntos

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

26657356

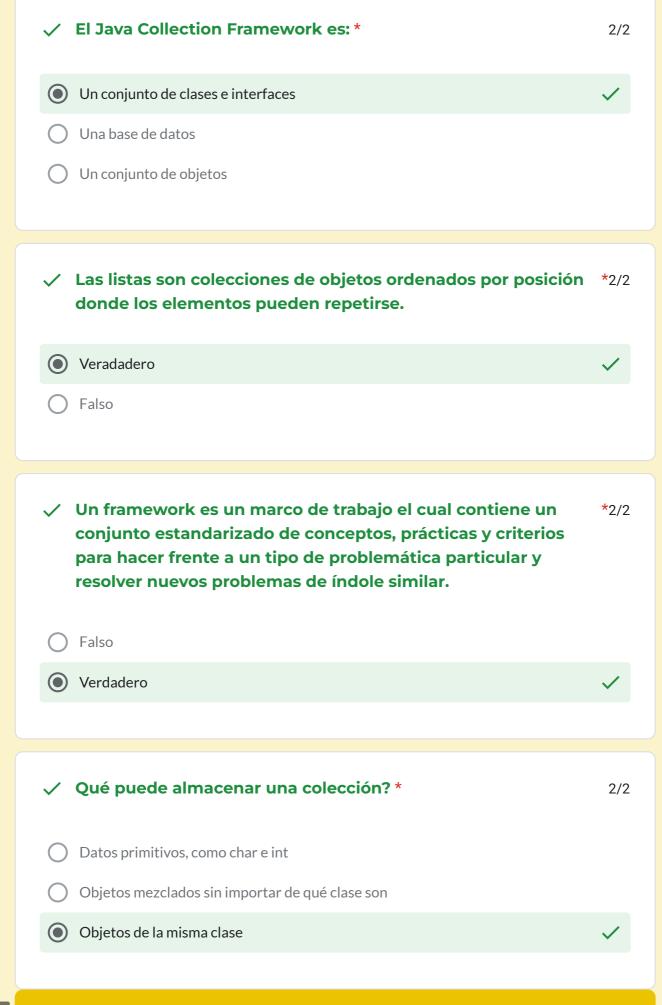
Correo Electrónico *

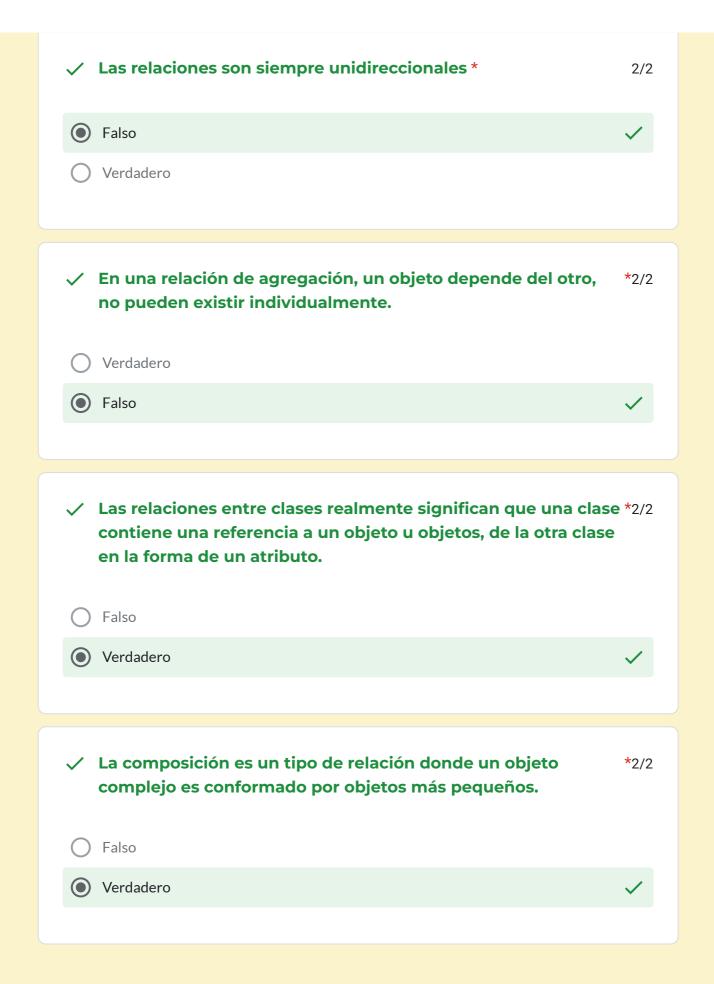
El mismo correo con el que Ingresas a tu Aula Virtual (con el que estás registrado en Egg)

spetrini@gmail.com

Nombre y Apellido *

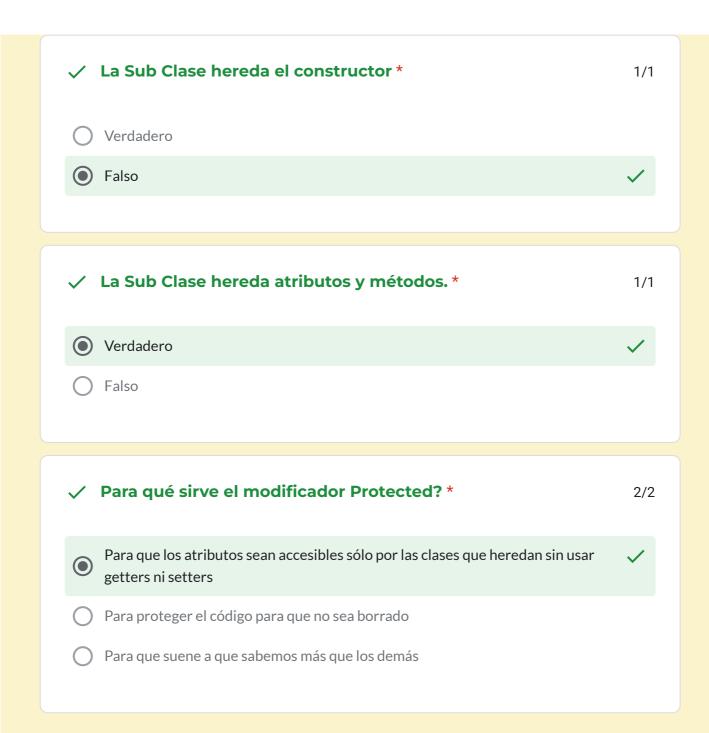
Sebastián A. Petrini





Las clases no actúan aisladas entre sí, al contrario, las clases están relacionadas unas con otras.	*2/2
Verdadero	✓
O Falso	
✓ Las relaciones pueden ser *	2/2
Uno a Uno y Muchos a Muchos	
O Uno a Uno solamente	
Uno a Uno, Uno a Muchos, Muchos a Muchos a Uno	✓
Herencia 15 de 15	puntos
✓ Cuándo usamos la palabra reservada super? *	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 cla padre	ase 🗸
Permite que la clase hija elimine métodos que considere innecesarios	
O Desbloquea nuevos métodos disponibles en Java	
✓ Una clase hija puede heredar de muchas clases padres. *	1/1
O Verdadero	
Falso	✓
✓ Una clase padre puede heredar a muchas clases hijas. *	2/2
Verdadero	✓
○ Falso	
✓ Qué es la Super Clase? *	2/2
O Una clase con superpoderes	
Como se llama a la clase superior(Padre o Madre)	✓
O Una clase con muchos métodos	

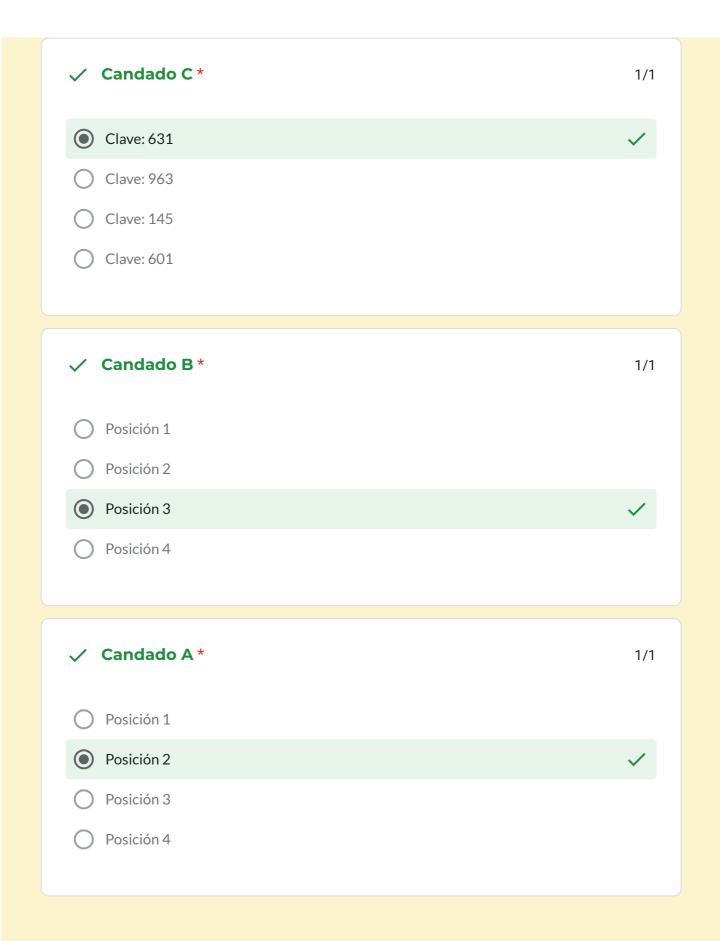


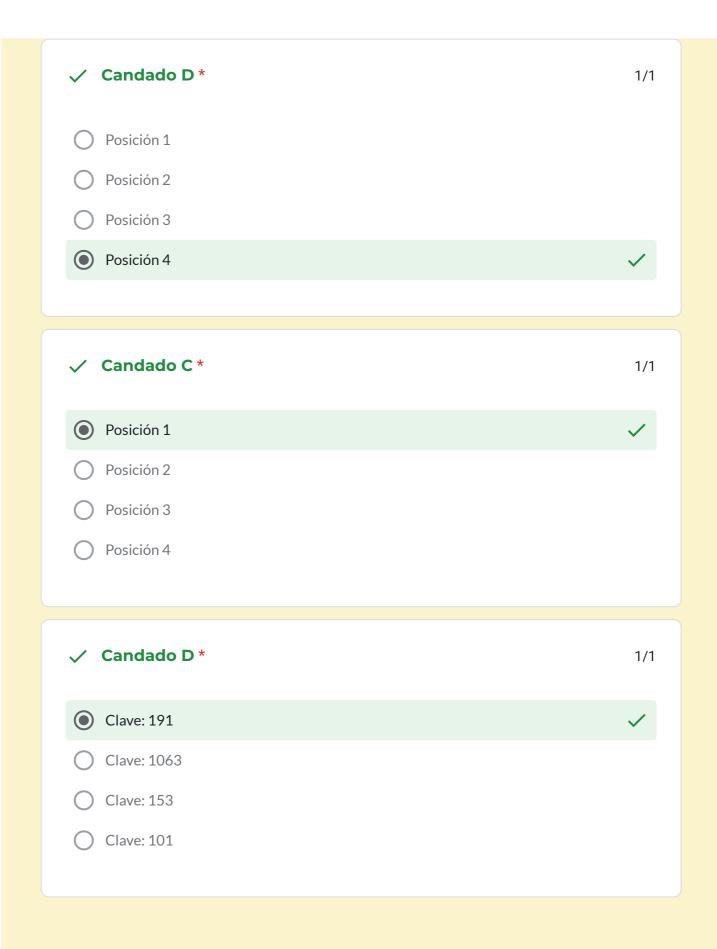
```
✓ En el siguiente código ¿cuál es la clase padre? *

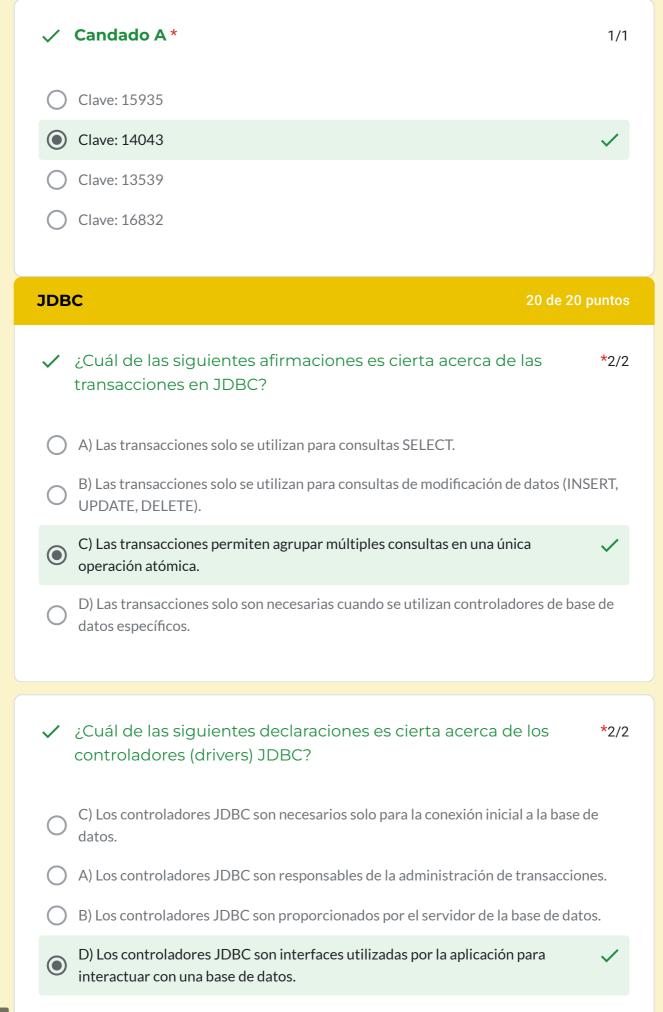
                                                                            3/3
      public class Persona extends SerVivo{
            String nombre;
      }
     Persona
     public class
     SerVivo
     Java.util
Manejo de Excepciones
✓ Cuál es el resultado del siguiente código? siguiendo el
                                                                           *4/4
    ejemplo que venimos viendo
     Persona p1 = new Persona();
    String nombre = null;
       if (!nombre.equals(null)) {
           p1.setNombre(nombre);
  8 } catch (Exception e) {
        System.out.println("El nombre no puede ser nulo");
    El nombre de p1 queda vacío
 Se imprime por consola "El nombre no puede ser nulo"
    El nombre de p1 es null
```



✓ 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
O Verdadero	
Falso	✓
MySQL 8 de 8	puntos
Resultados del ejercicio de los candados	
✓ Candado B *	1/1
✓ Candado B * Clave: 4380	1/1
	1/1
Clave: 4380	1/1







	¿Cuál es la forma más segura de ejecutar consultas SQL en JDBC para prevenir ataques de inyección de SQL?	*2/2
0	D) Utilizar el objeto ResultSet para obtener resultados de consultas.	
0	B) Utilizar el objeto Statement para ejecutar consultas sin parámetros.	
0	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.	✓
0	C) Mapear objetos a tablas de base de datos.	
0	D) Realizar operaciones CRUD en una base de datos.	
0	A) Crear consultas SQL.	
~	¿Cuál es el propósito del objeto ResultSet en JDBC?	2/2
0	C) Ejecuta una consulta SQL.	
0	D) Define los parámetros de una consulta parametrizada.	
•	B) Almacena el resultado de una consulta SQL.	✓
0	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.
O) 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
0	B) Utilizar un bucle y ejecutar consultas INSERT individuales.	
0	A) Ejecutar una consulta INSERT para cada fila a insertar.	
0	D) Utilizar el objeto BatchStatement para ejecutar consultas en lote.	
	C) Utilizar un objeto PreparedStatement y una transacción.	✓
✓ 8	¿Cuál es la forma correcta de manejar excepciones en JDBC? *	2/2
	C) Capturar excepciones específicas de JDBC y manejarlas adecuadamente.	✓
0	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.	
0	B) Lanzar una nueva excepción personalizada en cada método de JDBC.	
JPA	15 de 15 p	untos
	¿Cuál de las siguientes afirmaciones describe mejor JPA (Java Persistence API)?	*3/3
0	D) Es un framework para el desarrollo de interfaces de usuario en Java.	
0	B) Es una biblioteca para el acceso a bases de datos NoSQL.	
	C) Es una especificación de Java para el mapeo objeto-relacional.	✓
0	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 un relación de uno a muchos en JPA?	a * 2/2
C) @OneToOne	
D) @ManyToMany	
A) @OneToMany	✓
B) @ManyToOne	
✓ ¿Cuál s el método que se encarga de buscar y devolver una Entidad?	*3/3
	*3/3
Entidad?	*3/3
Entidad? output persist()	*3/3

Este formulario se creó en Egg Cooperation.

Google Formularios