

Integrador Java + MySQL

Total de puntos	3
-----------------	---

83/100



Co	rreo	مام	ctrói	aico	*
	rreo	ele	CILO	nico	•

jhonpaco@frba.utn.edu.ar

0 de 0 puntos

DNI*

Escribe tu cédula de identidad sin puntos, ni espacios

93913603

Nombre y Apellido *

Jhon Daniel Olmedo Paco

Correo Electrónico *

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

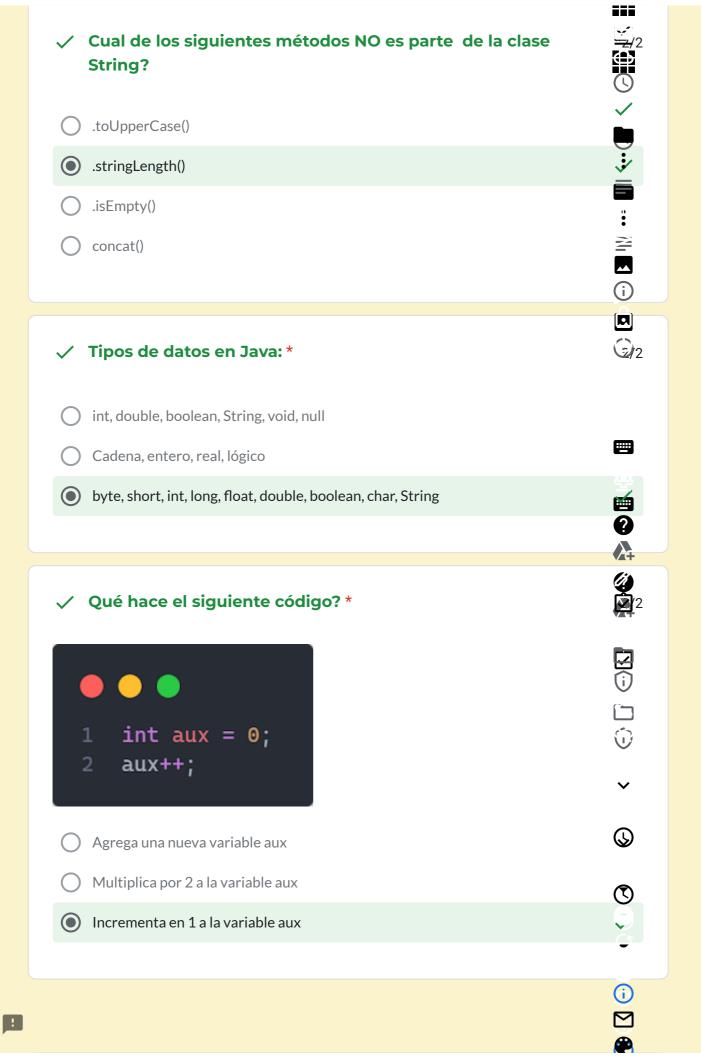
jhonpaco@frba.utn.edu.ar

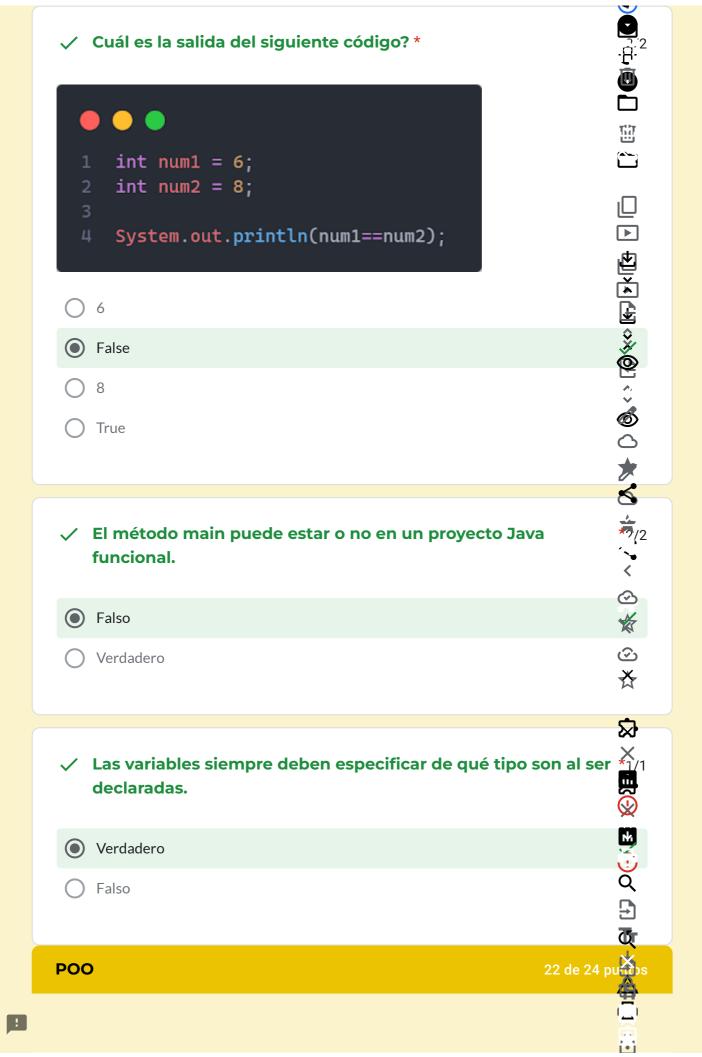
Java Introducción

11 de 11 puntos











Cuál de estos servicios trabaja de la manera correcta? *

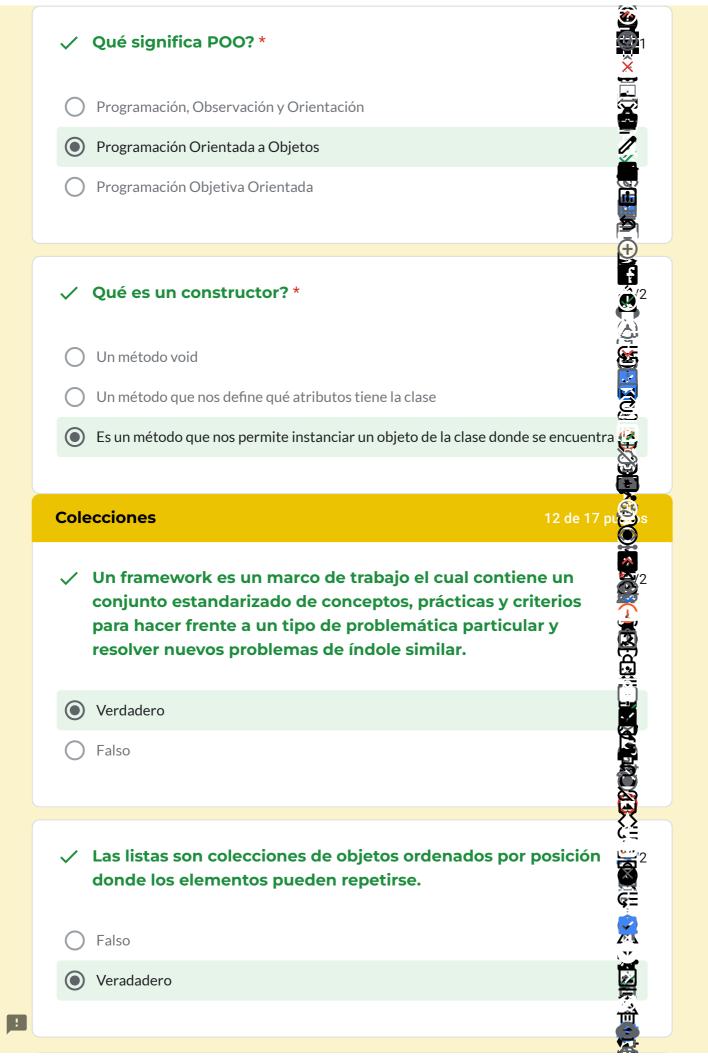


```
public void conocerNombre(Persona x){
    System.out.println("La persona se llama " + x.getNombre());
public void conocerNombre(String nombre){
    System.out.println("La persona se llama " + nombre);
```

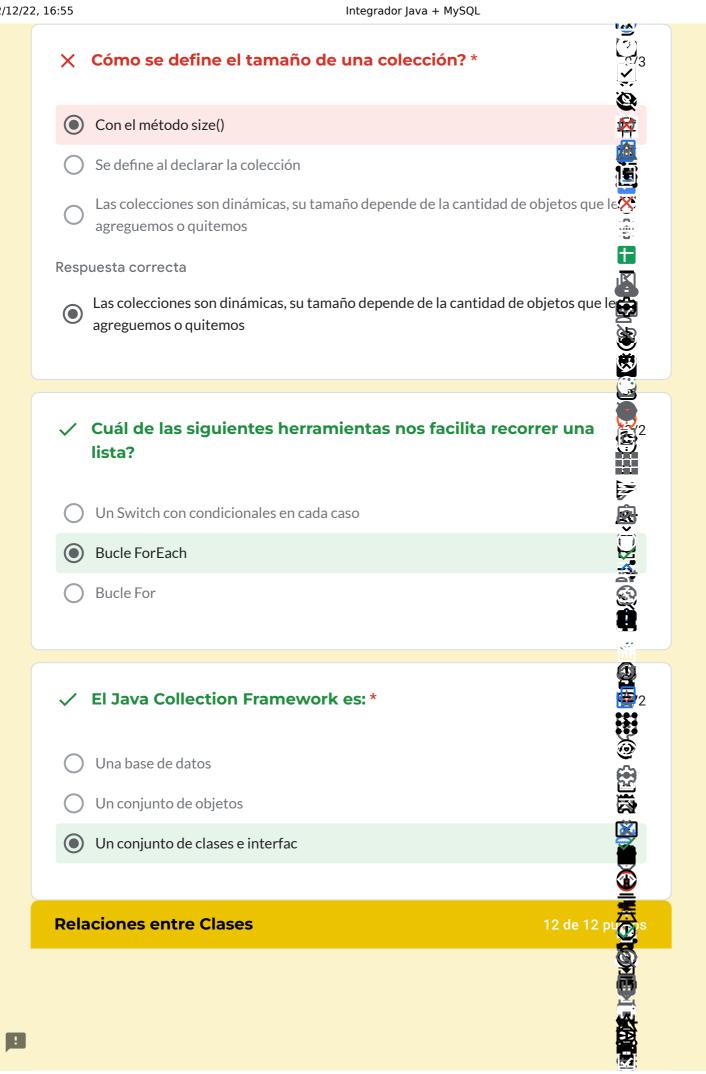
- El que obtiene el nombre desde un String
- El que obtiene el nombre a través de un objeto Persona
- Cuál es la diferencia entre clases y objetos? *
- Las clases son "moldes" y los objetos son modelados a partir de esos moldes, también llamados "instancias"
- Los objetos son abstracciones del mundo real, las clases son de lunes a jueves
- Las clases y los objetos son los mismo
- ¿De que se encarga Clase servicio o control? *
- Se encarga de crear y modificar objetos, teniendo cada clase de entidad una clase control.
- Se encarga de controlar que el programa no consuma memoria al instanciar objeto
- Se encarga de recibir los datos que ingresa el usuario por teclado







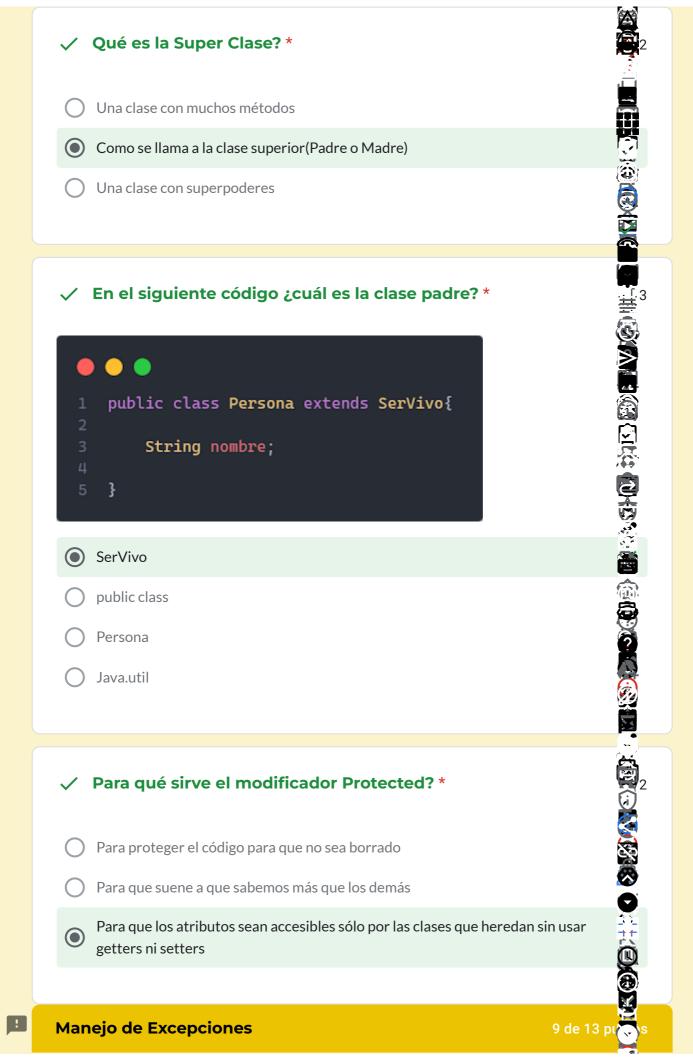
✓	Los mapas son colecciones de Llave/Valor. Las llaves son únicas pero los valores sí pueden repetirse.	
0	Falso	
•	Verdadero	
		\$
✓	Los conjuntos o Sets pueden contener elementos repetidos, simplemente se duplica el valor	
•	Falso	
0	Verdadero	
×	Qué puede almacenar una colección? *	2 2 2 3
0	Objetos de la misma clase	
•	Objetos mezclados sin importar de qué clase son	
0	Datos primitivos, como char e int	
Resp	uesta correcta	
•	Objetos de la misma clase	
		9



La composición es un tipo de relación donde un objeto complejo es conformado por objetos más pequeños.	
○ Falso	
Verdadero	
	(A)
	2
✓ Las relaciones pueden ser *	
Uno a Uno, Cero a Uno, Uno a Muchos y Cero a Muchos	
Uno a Uno y Muchos a Muchos	
O Uno a Uno solamente	
	Ž
	(3)
✓ En una relación de agregación, un objeto depende del otro,	
no pueden existir individualmente.	市
Falso	
Verdadero	
	1
	ij
	画
✓ Las clases no actúan aisladas entre sí, al contrario, las clases	
están relacionadas unas con otras.	TOP OF THE RE
Falso	\ <u>\</u>
Verdadero	

✓	Las relaciones son siempre unidireccionales *	2 2 8
•	Falso	1/1 1/2
0	Verdadero	
✓	Las relaciones son siempre unidireccionales * Falso Verdadero 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. Falso Verdadero	
0	Falso	
•	Verdadero	E E
Here	encia 15 de 15 p	s
✓	Una clase hija puede heredar de muchas clases padres. *	
\bigcirc	Verdadero	
•	Falso	
✓	Una clase padre puede heredar a muchas clases hijas. *	1 (2) (2) (3) (4) (4) (4) (4) (4) (4) (4) (4) (4) (4
•	Verdadero	2
0	Falso	

Permite que la clase hija elimine métodos que considere innecesarios Permite que la clase hija haga funcionar un método de manera diferente a la clase padre Desbloquea nuevos métodos disponibles en Java La Sub Clase hereda atributos y métodos.* Verdadero Falso La Sub Clase hereda el constructor * Verdadero Falso Cuándo usamos la palabra reservada super? * 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? *	
Desbloquea nuevos métodos disponibles en Java ✓ La Sub Clase hereda atributos y métodos. * ⑥ Verdadero Falso ✓ La Sub Clase hereda el constructor * ○ Verdadero ✓ Cuándo usamos la palabra reservada super? *	\bigcirc	Permite que la clase hija elimine métodos que considere innecesarios	
 ✓ La Sub Clase hereda atributos y métodos.* ✓ Verdadero ✓ Falso ✓ La Sub Clase hereda el constructor * ✓ Verdadero ✓ Falso ✓ Cuándo usamos la palabra reservada super? * 	•		
 ✓ La Sub Clase hereda atributos y métodos. * ● Verdadero → Falso ✓ La Sub Clase hereda el constructor * ● Verdadero ● Falso ✓ Cuándo usamos la palabra reservada super? * 	0	Desbloquea nuevos métodos disponibles en Java	
 ✓ La Sub Clase hereda el constructor * ○ Verdadero ⓒ Falso ✓ Cuándo usamos la palabra reservada super? * 	✓	La Sub Clase hereda atributos y métodos. *	
 ✓ La Sub Clase hereda el constructor * ○ Verdadero ⓒ Falso ✓ Cuándo usamos la palabra reservada super? * 	•	Verdadero	
 Verdadero Falso ✓ Cuándo usamos la palabra reservada super? * 	0	Falso	
 Verdadero Falso ✓ Cuándo usamos la palabra reservada super? * 			Š
✓ Falso✓ Cuándo usamos la palabra reservada super? *	✓	La Sub Clase hereda el constructor *	
✓ Cuándo usamos la palabra reservada super? *	0	Verdadero	
٠٠٠ - ١٠٠٠ - ١٠٠٠ - ١٠٠٠ - ١٠٠٠ - ١٠٠٠ - ١٠٠٠ - ١٠٠٠ - ١٠٠٠ - ١٠٠٠ - ١٠٠٠ - ١٠٠٠ - ١٠٠٠ - ١٠٠٠ - ١٠٠٠ - ١٠٠٠ -	•	Falso	
٠٠٠ - ١٠٠٠ - ١٠٠٠ - ١٠٠٠ - ١٠٠٠ - ١٠٠٠ - ١٠٠٠ - ١٠٠٠ - ١٠٠٠ - ١٠٠٠ - ١٠٠٠ - ١٠٠٠ - ١٠٠٠ - ١٠٠٠ - ١٠٠٠ - ١٠٠٠ -			
m.			豐
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	✓	Cuándo usamos la palabra reservada super? *	
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			Ä
Cuando queremos acceder a los métodos, atributos y constructores de la clase superior Cuando queremos referirnos a una clase que está muy bien hecha	0		
Cuando queremos referirnos a una clase que está muy bien hecha	•		
	0	Cuando queremos referirnos a una clase que está muy bien hecha	S S S S S S S S S S S S S S S S S S S



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

```
Persona p1 = new Persona();

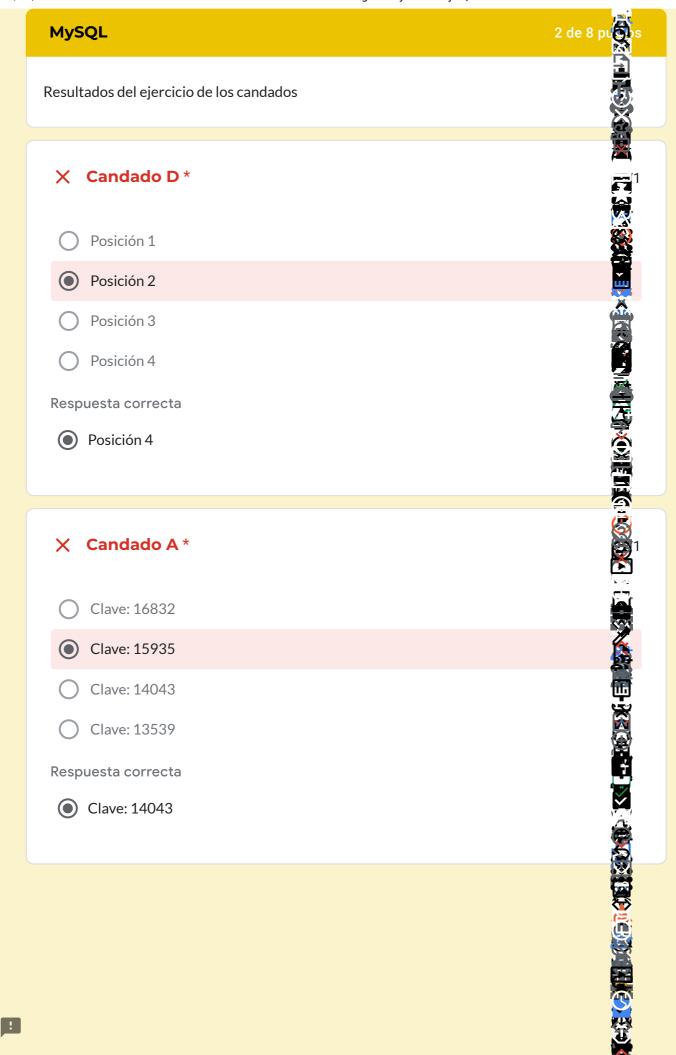
String nombre = null;

try {
    if (!nombre.equals(null)) {
       p1.setNombre(nombre);
    }

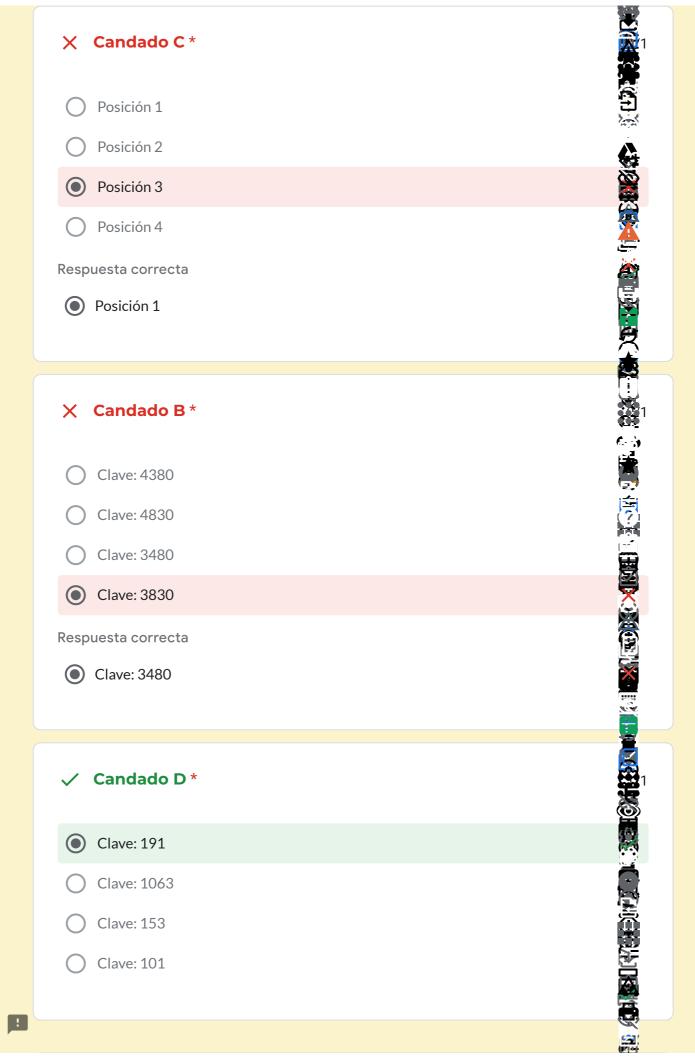
catch (Exception e) {
    System.out.println("El nombre no puede ser nulo");
}
```

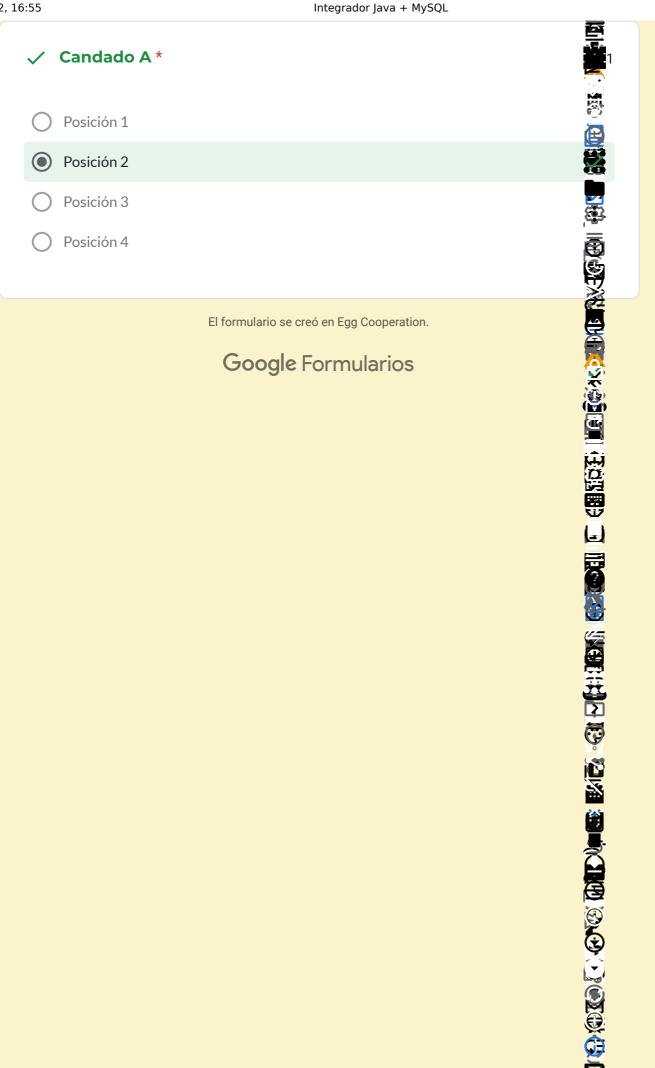
- Se imprime por consola "El nombre no puede ser nulo"
- El nombre de p1 queda vacío
- El nombre de p1 es null
- Un objeto Exception contiene información sobre un error ocurrido.
- Verdadero
- Falso
- ✓ Una excepción es un evento que ocurre durante la ejecución de un programa que interrumpe el flujo normal de las instrucciones del programa.
- Verdadero
- Falso

	×	Sólo se puede usar 1 bloque catch por bloque try. *	1 2
	•	Verdadero	
	\bigcirc	Falso	
	Resp	uesta correcta	
		Falso	A ±
	✓	Para aclarar que un método puede lanzar un error se coloca la palabra "extends".	
	0	Verdadero	
		Falso	
	×	Si una excepción se produce en el bloque catch, puede atraparse en el bloque try y manejarla. Falso	2 6 6 6 6
		Verdadero	
	Resp	uesta correcta	
	•	Falso	
	✓	Cuando se lanza una excepción no hay nada más que hacer, termina el programa.	1
	0	Verdadero	
		Falso	
!			



★ Candado B *	(A) 1
O Posición 1	
O Posición 2	
O Posición 3	9
Posición 4	
Respuesta correcta	
Posición 3	
X Candado C *	
O Clave: 631	
O Clave: 963	
Clave: 145	
O Clave: 601	
Respuesta correcta	
Clave: 631	











12/12/22, 16:55	Integrador Java + MySQL