

1. Que es una excepción

- a) Condición que puede saltar una regla
- b) Un error de sintaxis

c) Cualquier error que ocurre durante la ejecución del programa y hace que se detenga

- d) Un error de java

2. En que se divide las excepciones

a) IOException (Comprobados) y RuntimeException (No comprobados)

- b) IOException (Error de hardware) y RuntimeException (Throwable)
- c) IOException (No comprobados) y RuntimeException (Comprobados)
- d) Throwable (Excepciones) y Error de hardware (Comprobados)

3. Como podemos definir una excepción comprobada

- a) Cuando el programa se detiene a la mitad por culpa de un error que no es necesario controlar

b) Cuando tenemos un error ajeno a nuestro programa que es necesario controlar

- c) Tengo un error lógico que el java no puede detectar
- d) Cuando tengo un error que ya se de antemano y es necesario controlar

4. Cuantas maneras existen de declarar un error

- a) Solo una, con un try and catch

b) Solo una, con un throw

- c) Hay dos maneras, con un throw o con un try and catch
- d) Solo una, con un throw and catch

5. Cual de los siguientes es un error comprobado

- a) Arithmetic Exception
- b) NullPointerException

c) FileNotFoundException

- d) Ninguna de las anteriores

6. Que es una excepción personalizada

a) Un error que crea el mismo usuario para su programa

- b) Un error que crea el Java para el programa del usuario
- c) Un error que declara un usuario para su programa
- d) Todas las anteriores

7.Cuál es el propósito de una excepción personalizada

- a) Para dejar de usar while y do while
- b) Porque hay un error no lógico que el java no lo detecta

c) Porque hay un error lógico que el java no lo detecta

- d) Porque no quiero que al final del programa existan errores

8. Cuál de las siguientes es la manera de crear una excepción personalizada

- a) `public class ExpecionPersonalizadaNombre extends IO exception{}`
- b) `private class ExpecionPersonalizadaNombre extends IO exception{}`

c) `public class ExpecionPersonalizadaNombre extends Exception{}`

- d) `private class ExpecionPersonalizadaNombre extends Exception{}`

9. De las siguientes opciones, que se puede hacer con una excepción personalizada

- a) Crear un error de hardware
- b) Crear mis propios mensajes de error**
- c) Declarar excepciones
- d) Crear una excepción comprobada

10. Donde se guardan los errores cuando capturo un error

- a) En el programa
- b) En el Java
- c) En una variable cualquiera**
- d) En una variable propia de Java