

Práctica 1. Documentación de 5 errores de código Java

Guerrero Garza Gerardo Omar

- **Error 1. Provocado porque quite el main de la declaración del método**

```
HolaMundo.java:7: error: <identifier> expected
    public static void (String[] args) {
                       ^
```

1 error

Forma correcta de arreglarlo con el main:

```
public static void main(String[] args) {
    // Código aquí
}
```

- **Error 2. Java indica que falta un punto y coma ; al final de una instrucción. (Se lo quite a un System.out.println)**

```
HolaMundo.java:5: error: ';' expected
    System.out.println("GNU/Linux (pronunciado oficialmente como ñu linux o también
ge-ene-u linux en español),2 es una familia de sistemas operativos tipo Unix compuesto por
software libre y de código abierto.3 GNU/Linux surge de las contribuciones de varios
proyectos de software, entre los cuales destacan GNU (iniciado por Richard Stallman en
1983) y el núcleo Linux (comenzado por Linus Torvalds en 1991).4")
```

^

1 error

- **Error 3. Java indica que el compilador llegó al final del archivo sin encontrar el cierre correcto de un bloque de código o una estructura de control. (Quite la última llave que cerraba a la public class HolaMundo)**

```
HolaMundo.java:14: error: reached end of file while parsing
    System.out.println("El nucleo del Sistema operativo gestiona todos los procesos
vitales y recursos. El nucleo en su mas pura forma es distante del usuario, su misión es a un
bajo nivel y muy dificilmente el usuario interactúa con el. Pero el nucleo en si no hace
alusión a ninguna interfaz gráfica ni programa que de manera básica interactue con el
usuario."); }
```

^

1 error

- **Error 4. Elimine los parámetros del método main contenidos en el paréntesis (String[] args) y por ello indica que el compilador no pudo encontrar el método main en mi clase HolaMundo**

Error: no se ha encontrado el método principal en la clase HolaMundo, defina el método principal del siguiente modo:\n public static void main(String[] args)\nde lo contrario, se deberá ampliar una clase de aplicación JavaFX javafx.application.Application

Error 5. Este error ocurre porque en Java, el nombre de una clase pública (public class) debe coincidir exactamente con el nombre del archivo que la contiene. Esto es una regla estricta en Java.

HolaMundo.java:2: error: class HolaMudo is public, should be declared in a file named HolaMudo.java

```
public class HolaMudo{
```

^

1 error