

Ejercicio 1:

Pasos para Instalar Visual Studio Code y Crear un Proyecto de Prueba en Java:

1. Instalar Visual Studio Code:

- Descarga e instala Visual Studio Code desde su sitio oficial.
- Sigue las instrucciones de instalación para tu sistema operativo.

2. Instalar Extensiones para Java:

- Abre Visual Studio Code.
- Ve a la pestaña de Extensiones (puedes hacer clic en el ícono de extensiones en la barra lateral izquierda o presionar Ctrl+Shift+X).
- Busca "Java Extension Pack" y "Language Support for Java" e instálalas.

3. Crear un Proyecto de Prueba:

- Abre Visual Studio Code y selecciona la opción "View" en la barra de menú, luego "Command Palette" o simplemente presiona Ctrl+Shift+P.
- Escribe "Java: Create Java Project" y selecciona esta opción.
- Sigue las instrucciones para dar un nombre al proyecto y seleccionar una ubicación.
- Selecciona el tipo de proyecto (por ejemplo, "Hello World").

4. Escribir un Programa Simple (Hola Mundo):

- Dentro de tu proyecto, encontrarás una carpeta llamada "src" (fuente). Dentro de esta carpeta, encontrarás el archivo principal de tu programa Java.
- Abre este archivo y escribe el siguiente código de "Hola Mundo":
java

```
public class HelloWorld {  
    public static void main(String[] args) {  
        System.out.println("Hola Mundo");  
    }  
}
```

-
- Guarda el archivo.

2. Ejecutar el Programa:

- Abre una terminal integrada en Visual Studio Code.
- Navega a la carpeta del proyecto y compila el programa usando el comando `javac HelloWorld.java`.
- Ejecuta el programa con el comando `java HelloWorld`. Deberías ver la salida "Hola Mundo".

Ejercicio 2:

Fases del Ciclo de Vida del Software:

El ciclo de vida del software se compone de varias fases, que pueden variar según el modelo de desarrollo utilizado. Un modelo común es el Modelo de Desarrollo en Cascada. Aquí están las fases generales:

1. **Requisitos:**
 - Identificación y documentación de los requisitos del software.
 - Comprender las necesidades y expectativas del cliente.
2. **Diseño:**
 - Creación de la arquitectura del sistema.
 - Diseño detallado de módulos y componentes.
 - Especificación de la interfaz del usuario.
3. **Implementación (Codificación):**
 - Traducción del diseño en código fuente.
 - Desarrollo de funciones y componentes.
 - Pruebas unitarias para asegurar que cada componente funcione correctamente.
4. **Pruebas:**
 - Verificación de que el software cumple con los requisitos especificados.
 - Identificación y corrección de errores.
 - Pruebas de integración para verificar la interacción entre los módulos.
5. **Despliegue:**
 - Lanzamiento del software en un entorno de producción.
 - Configuración y ajuste según sea necesario.
6. **Mantenimiento:**
 - Corrección de errores y problemas.
 - Actualización del software para satisfacer nuevas necesidades o requisitos.