

Práctica Guiada: Trabajando con Archivos JAR en Java

Introducción a JAR:

Esta práctica va sobre la creación, visualización, actualización, extracción y verificación de archivos JAR, utilizando IntelliJ IDEA como IDE.

Conceptos Básicos:

- Archivo JAR (Java Archive): Un formato de archivo que permite empaquetar y comprimir archivos Java, como clases, recursos y metadatos, en un solo archivo.
- Bytecode Java: Código intermedio ejecutable por la JVM. Las clases Java se compilan en bytecode antes de ejecutarse.
- Manifiesto: Archivo dentro del JAR que contiene metadatos sobre la aplicación, como la versión y la clase principal.

Pasos de la Práctica:

Paso 1: Crear un Proyecto en IntelliJ IDEA

1. Abre IntelliJ IDEA y selecciona "File" -> "New" -> "Project".
2. Selecciona "Java" como tipo de proyecto y haz clic en "Next".
3. Completa los detalles del proyecto y haz clic en "Finish".

Paso 2: Crear Clases Java

1. Crea una clase Java llamada `MiApp` con un método `main`:

```
```java
public class MiApp {
 public static void main(String[] args) {
 System.out.println("¡Bienvenido a MiApp!");
 }
}
```
```

Paso 3: Compilar las Clases

1. Desde IntelliJ, selecciona "Build" -> "Build Project". Esto compilará las clases Java y generará archivos `.class`.

Paso 4: Crear un Archivo Manifiesto

1. Crea un archivo de texto llamado `Manifiesto.txt` en la misma ubicación que tu clase `MiApp`.
2. Agrega el siguiente contenido al archivo `Manifiesto.txt`:

```
```
```

```
Manifest-Version: 1.0
Main-Class: MiApp
...
```

#### Paso 5: Crear un Archivo JAR Manualmente

1. Abre la terminal en la ubicación de tu proyecto y ejecuta el siguiente comando para crear el archivo JAR:

```
...
jar cfm MiApp.jar Manifesto.txt MiApp.class
...
```

Este comando empaqueta el archivo manifiesto y la clase `MiApp` en un archivo JAR llamado `MiApp.jar`.

#### Paso 6 (Adicional): Visualizar Contenido del Archivo JAR

1. Usa el comando `jar tf MiApp.jar` en la terminal para ver el contenido del archivo JAR.

```
...
jar tf MiApp.jar
...
```

#### Paso 7 (Adicional): Actualizar el Archivo JAR

1. Actualiza la clase `MiApp` con un nuevo mensaje.

```
...
public class MiApp {
 public static void main(String[] args) {
 System.out.println("¡Actualizado a una nueva versión de MiApp!");
 }
}
...
```

2. Recompila el proyecto desde IntelliJ IDEA.

3. Usa el siguiente comando para actualizar el contenido del archivo JAR:

```
...
jar uf MiApp.jar MiApp.class
...
```

#### Paso 8 (Adicional): Extraer Archivos del JAR

1. Usa el siguiente comando para extraer la clase `MiApp.class` del JAR:

```
...
jar xf MiApp.jar MiApp.class
...
```

Este comando extrae la clase del JAR y la coloca en el directorio actual.