

4. Practical Exercise: Build and Run a Java Application with Maven, Migrate the Same Application to Gradle

Solution:

This exercise will guide you through building a basic Java application using Maven and then demonstrate the steps to migrate the same application to a Gradle build.

Create Maven Project in IntelliJ

1. Open **IntelliJ IDEA**
2. Go to: **File** → **New** → **Project**
3. Choose: **Maven**
4. GroupId: `com.example`
5. ArtifactId: `demoapp`
6. Check: `Create from archetype` → Select `maven-archetype-quickstart`
7. Finish.

Run the Application

Open `App.java` at `src/main/java/com/example/App.java`

Add a simple `main()` method if not already present:

```
package com.example;
```

```
public class App {  
    public static void main(String[] args) {  
        System.out.println("Hello from Maven application!");  
    }  
}
```

Right-click on the file → **Run 'App.main()'**

PART 2: Migrate to Gradle Using **gradle init**

1 Open Terminal in IntelliJ (Bottom Panel)

Navigate to the Maven project root, then run:

```
gradle init
```

2 Choose Initialization Options

When prompted:

- Type of project: **application**
- Language: **Java**
- DSL: **Groovy**
- Test Framework: **JUnit**
- Source package: **com.example**

It creates a **build.gradle**, **settings.gradle**, and **src** folder.

3 Replace Gradle Source Code

1. Move your existing **App.java** to Gradle's structure:
2. Delete the Maven folder if necessary.

4 Sync Gradle in IntelliJ

- IntelliJ will detect the **build.gradle**
- Click **Import Gradle Project** or **Load Gradle Changes**

5 Run with Gradle

- Right-click `App.java` → **Run 'App.main()'**
- Or open the **Gradle Tool Window** → `Tasks > application > run`