Step 3: Model Creation

In this step, you'll create a Java class to represent a high score entity. This class will act as the model in your application's MVC (Model-View-Controller) architecture.

Steps to Create the Model

- **Create New Package**
  - In the **src/main/java/com/example/highscoreapi** directory, create a new package and name it **model**.
- **Create HighScore Class**
  - Inside the **model** package, create a new Java class and name it **HighScore**.
- **Add Annotations and Fields**
  - Annotate the class with **@Entity** to indicate that it's a JPA entity.
  - Create fields for **id**, **playerName**, and **score**.
  - Annotate **id** with **@Id** and **@GeneratedValue(strategy = GenerationType.IDENTITY)** for auto-generation.

Here is how your **HighScore** class should look:

java

Copy code

```java
package com.example.highscore.model;

import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;

@Entity
public class HighScore {

    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Long id;
    private String playerName;
    private int score;
```

```
    // Getters and Setters
}
```

Documentation

Update the **README.md** with details of this step:

markdown

Copy code

```
## Step 3: Model Creation

We created a `HighScore` class inside a new package named `model`. This class serves as the JPA entity and contains fields for `id`, `playerName`, and `score`. Annotations are used to specify the `id` as the primary key and to enable auto-generation for it.
```

Commit Changes

After completing this step, it's a good idea to commit these changes to your Git repository.

- Run **git add .** to stage the new files and changes.
- Run **git commit -m "Created HighScore model"** to commit the staged changes.

Conclusion

You've successfully defined the **HighScore** class to represent the data model in your application. It has the necessary JPA annotations to map it to a database table.