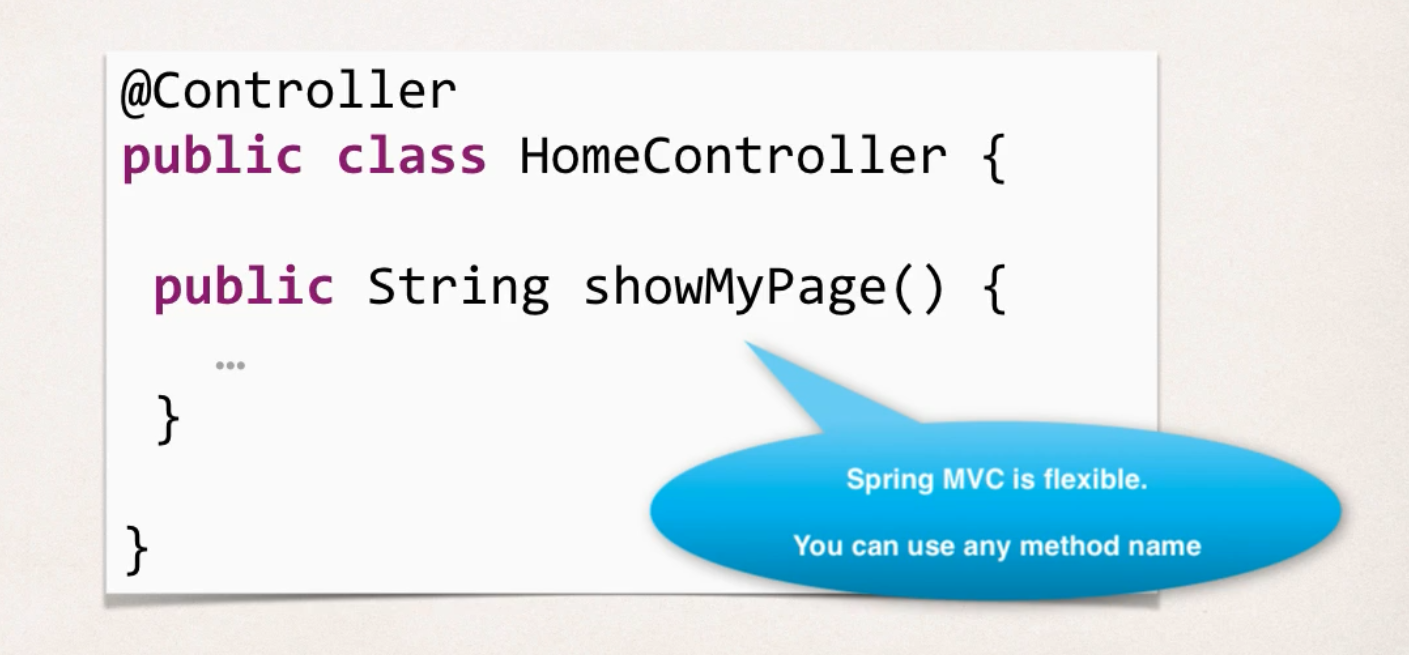
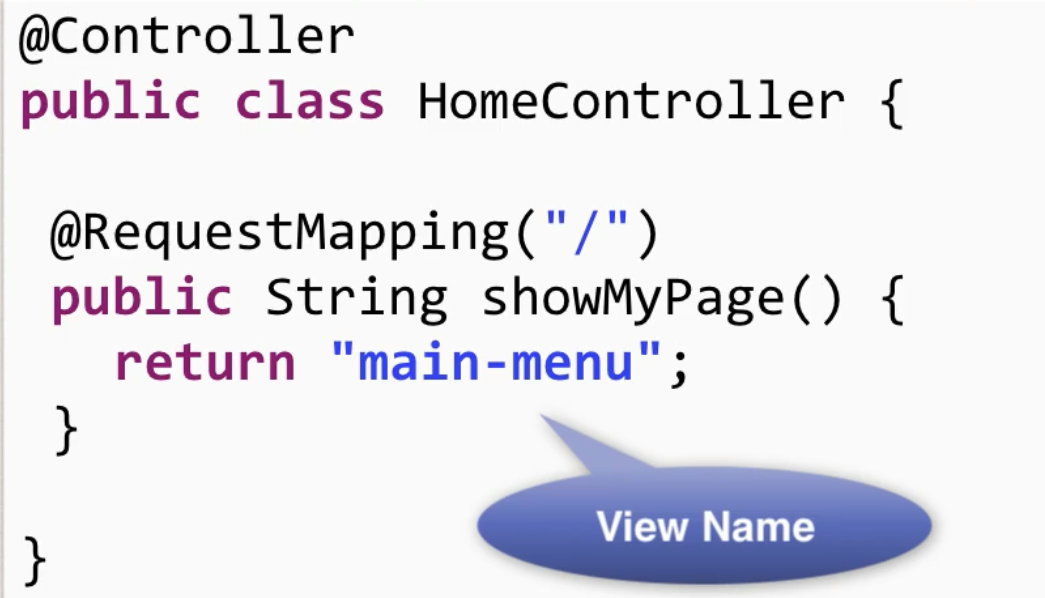
# Controller

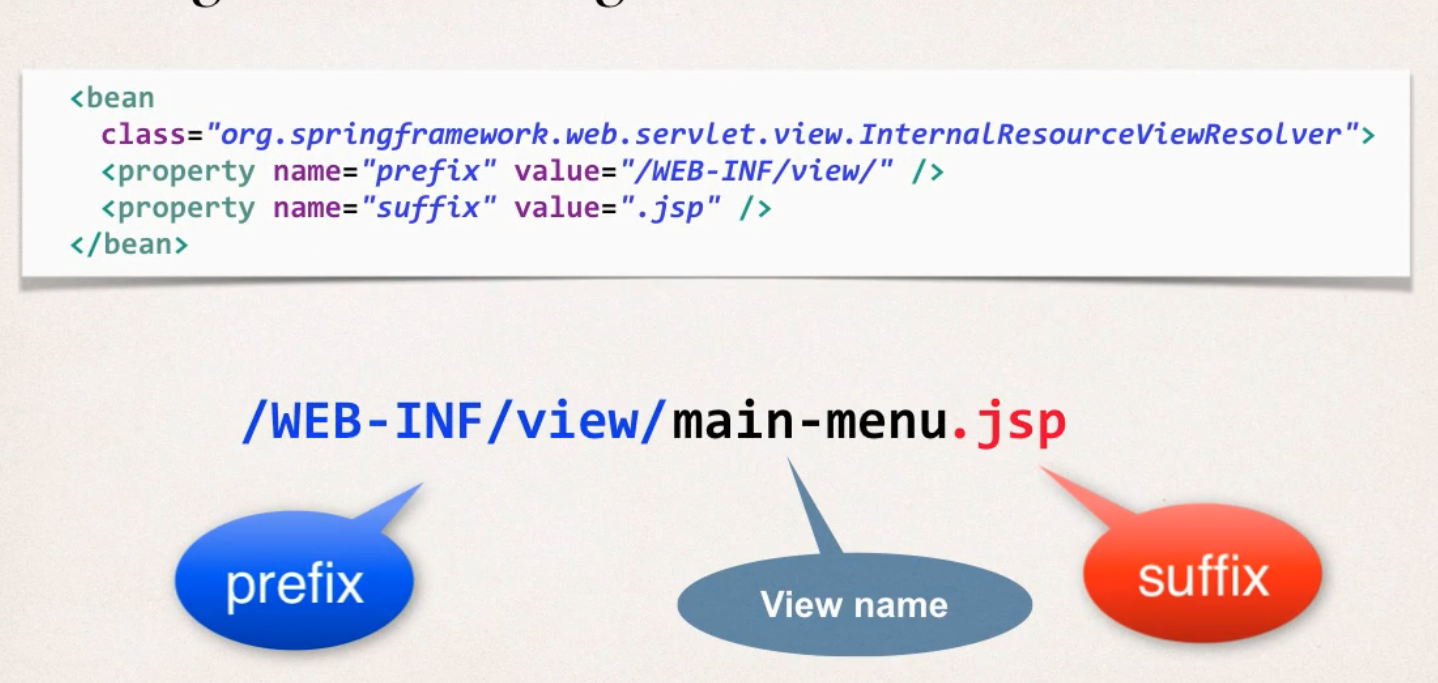
1. **Creating Controller Class:** Basically, we have to create a class with **@Controller** annotation.
   * @Controller inherits from @Component, so it’s just a specialized component that supports web-mvc.
2. **Define Controller method:** This method will act as our controller.
   * We can pass any number of parameters like request, response, session objects, model objects etc.
   * We can also return different types.



1. **Add request mapping to Controller method:** Here we add mapping to some kind of web requestion to this method.
   * It is done with the annotation **@RequestMapping(“/”)**. It defines the actual path that we want to map.



1. **Return view Name:** Here we return the view name that is to be shown.
   * Here, Spring will resolve the view name according to our configuration (step 5 of our setup process),



1. **Develop view Page:** This any regular html/jsp page.

If we follow all these steps, we should be able to run the app. If any configuration error happens, check for the imports in the xml files and the paths.

* There are multiple reasons for that EL (JSP Expression Language) won’t work, it can be that Tomcat or servlet version is below the minimum requirement, or the IDE don’t recognize it. In this case we need to manually enable.



Using parameters in JSX page using EL: Syntax is similar to the template string in JavaScript.

