How I approached Task 1
HelloWorld.java

Kushagra Raj Tiwari

START

1. Reading Spring Boot Docs.
2. Installing Eclipse IDE
3. Installing Spring Tool Suite in eclipse
4. Creating New Spring Starter Project **HelloWorld** with Maven as Build Tool using JDK 1.8, Java as a Language and dependency as Spring Web.

Technologies Used

1. Google Chrome
2. Eclipse 2020-03 IDE
3. Java JDK 1.8
4. Spring Boot 2.2.6
5. Apache Maven 3.6.3
6. Spring Tool Suite
7. Spring Web 2.2.6 Dependency
8. Git

1. Creating New class **Control (@RestController)** for handling endpoint web requests. It will have a method **String show() (@RequestMapping("/hello"))**
2. **To show loose coupling in Spring**, creating new class **Message (@Component)**. Spring will treat this class as spring bean and create its object at runtime. This class wil have a public method String **showMessage()** which will return "Hello! World" message.

1. Accessing object (**not** creating tightly coupled with **new**) of class **Message** with **@Autowired** already created by spring in class **Contro**l.
2. This now this object is used in show() method to call showMessage() which will return our message.
3. Finally going to HelloWorld.java and running application as Spring Boot App on Tomcat.
4. Running http://localhost:8080/hello . will show "Hello! World" message.
5. Shutdown HelloWorld.java
6. Created Github Repository for project.

END