***How Spring boot initialize the Spring MVC application context?***

When Springboot is bootstrap using the below code it loads the Spring MVC configuration automatically.

package hello;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

public class Application {

public static void main(String[] args) {

SpringApplication.run(Application.class, args);

}

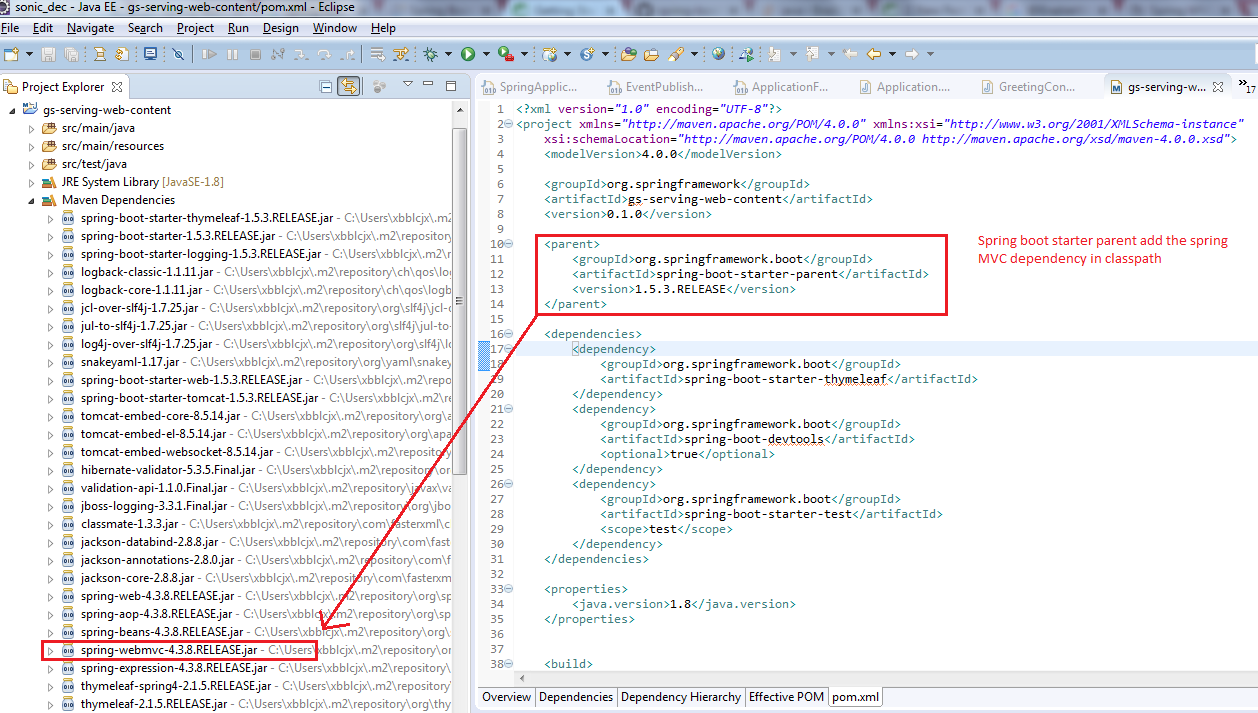
}

Under the hood of @SpringBootApplication when the above code is executed it adds the below annotation.

1. @Configuration tags the class as a source of bean definitions for the application context.
2. @EnableAutoConfiguration tells Spring Boot to start adding beans based on classpath settings, other beans, and various property settings.

Normally you would add @EnableWebMvc for a Spring MVC app (explicit declaration ), but Spring Boot adds it automatically when it sees **spring-webmvc** on the classpath. This flags the application as a web application and activates key behaviors such as setting up a DispatcherServlet.

@ComponentScan tells Spring to look for other components, configurations, and services in the hello package, allowing it to find the controllers.



Different option to initialize the Spring MVC context if NOT using spring boot.

1. @EnableWebMvc + extending WebMvcConfigurerAdapter  and customizing some of the default configuration.
   * With this mechanism you are basically doing some autoconfiguration provided by @EnableWebMvc
2. Just extends WebMvcConfigurationSupport and customize the default configuration

If using spring boot

1. Add @EnableAutoConfiguration annotation and extends WebMvcConfigurerAdapter. No need to use ~~@EnableMvc~~

So option 3 is more generic .

Difference between **WebMvcConfigurationSupport and WebMvcConfigurerAdapter**

|  |  |
| --- | --- |
| **WebMvcConfigurationSupport** | **WebMvcConfigurerAdapter** |
| In the Spring MVC framework, this is the main class providing the configuration behind the MVC Java config. | This is simple adapter class for customizing some of the default configuration. |
| We could use the @EnableWebMvc annotation to import the configuration of this class automatically. | This is not related to the @EnableWebMvc annotation. |
| We could extend WebMvcConfigurationSupport class to prevent the default configuration imported by the @EnableWebMvc annotation. | We could extend WebMvcConfigurerAdapter class to customize some of the default configuration imported by @EnableWebMvc annotation. |
| In the java configuration file for spring mvc either we could use @EnableWebMvc annotation or extend WebMvcConfigurationSupport class. | If we are extending this class, still required to use @EnableWebMvc annotation, otherwise application will not work. |

So Springboot initialize WebMvcConfigurationSupport for you.If you see the Java API doc. It done the following –

This is the main class providing the configuration behind the MVC Java config. It is typically imported by adding [@EnableWebMvc](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/servlet/config/annotation/EnableWebMvc.html) to an application [@Configuration](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/context/annotation/Configuration.html) class. An alternative more advanced option is to extend directly from this class and override methods as necessary remembering to add [@Configuration](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/context/annotation/Configuration.html) to the subclass and [@Bean](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/context/annotation/Bean.html) to overridden [@Bean](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/context/annotation/Bean.html) methods. For more details see the Javadoc of [@EnableWebMvc](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/servlet/config/annotation/EnableWebMvc.html).

This class registers the following [HandlerMapping](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/servlet/HandlerMapping.html)s:

* [RequestMappingHandlerMapping](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/servlet/mvc/method/annotation/RequestMappingHandlerMapping.html) ordered at 0 for mapping requests to annotated controller methods.
* [HandlerMapping](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/servlet/HandlerMapping.html) ordered at 1 to map URL paths directly to view names.
* [BeanNameUrlHandlerMapping](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/servlet/handler/BeanNameUrlHandlerMapping.html) ordered at 2 to map URL paths to controller bean names.
* [HandlerMapping](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/servlet/HandlerMapping.html) ordered at Integer.MAX\_VALUE-1 to serve static resource requests.
* [HandlerMapping](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/servlet/HandlerMapping.html) ordered at Integer.MAX\_VALUE to forward requests to the default servlet.

Registers these [HandlerAdapter](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/servlet/HandlerAdapter.html)s:

* [RequestMappingHandlerAdapter](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/servlet/mvc/method/annotation/RequestMappingHandlerAdapter.html) for processing requests with annotated controller methods.
* [HttpRequestHandlerAdapter](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/servlet/mvc/HttpRequestHandlerAdapter.html) for processing requests with [HttpRequestHandler](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/HttpRequestHandler.html)s.
* [SimpleControllerHandlerAdapter](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/servlet/mvc/SimpleControllerHandlerAdapter.html) for processing requests with interface-based [Controller](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/servlet/mvc/Controller.html)s.

Registers a [HandlerExceptionResolverComposite](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/servlet/handler/HandlerExceptionResolverComposite.html) with this chain of exception resolvers:

* [ExceptionHandlerExceptionResolver](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/servlet/mvc/method/annotation/ExceptionHandlerExceptionResolver.html) for handling exceptions through @[ExceptionHandler](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/bind/annotation/ExceptionHandler.html) methods.
* [ResponseStatusExceptionResolver](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/servlet/mvc/annotation/ResponseStatusExceptionResolver.html) for exceptions annotated with @[ResponseStatus](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/bind/annotation/ResponseStatus.html).
* [DefaultHandlerExceptionResolver](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/servlet/mvc/support/DefaultHandlerExceptionResolver.html) for resolving known Spring exception types

Registers an [AntPathMatcher](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/util/AntPathMatcher.html) and a [UrlPathHelper](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/util/UrlPathHelper.html) to be used by:

* the [RequestMappingHandlerMapping](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/servlet/mvc/method/annotation/RequestMappingHandlerMapping.html),
* the [HandlerMapping](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/servlet/HandlerMapping.html) for ViewControllers
* and the [HandlerMapping](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/servlet/HandlerMapping.html) for serving resources

Note that those beans can be configured with a [PathMatchConfigurer](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/servlet/config/annotation/PathMatchConfigurer.html).

Both the [RequestMappingHandlerAdapter](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/servlet/mvc/method/annotation/RequestMappingHandlerAdapter.html) and the [ExceptionHandlerExceptionResolver](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/servlet/mvc/method/annotation/ExceptionHandlerExceptionResolver.html) are configured with default instances of the following by default:

* a [ContentNegotiationManager](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/accept/ContentNegotiationManager.html)
* a [DefaultFormattingConversionService](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/format/support/DefaultFormattingConversionService.html)
* a [OptionalValidatorFactoryBean](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/validation/beanvalidation/OptionalValidatorFactoryBean.html) if a JSR-303 implementation is available on the classpath
* a range of [HttpMessageConverter](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/http/converter/HttpMessageConverter.html)s depending on the third-party libraries available on the classpath.

The flow diagram looks like below

Springboot application is Bootstraped SpringApplication.run(Application.class, args);

Look for spring-mvc libraries in classpath

Imports the auto configuration(@EnableWebMvc

Intialize

WebMvcConfigurationSupport start

Registers the [HandlerMapping](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/servlet/HandlerMapping.html)s

Registers the [HandlerAdapter](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/servlet/HandlerAdapter.html)s

Registers a [HandlerExceptionResolverComposite](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/servlet/handler/HandlerExceptionResolverComposite.html)

Registers an [AntPathMatcher](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/util/AntPathMatcher.html) and a [UrlPathHelper](https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/util/UrlPathHelper.html)

Intialize

WebMvcConfigurationSupport end

Intialize

WebMvcConfigurationSupport start