What problem does Spring Boot solve?

* Eliminate the pain of the start of dependencies and also to provide auto configuration by automatically registering their spring beans that you would have to do after you would have gone to get those dependencies.

@SpringBootApplication:

It basically defines 3 Annotations here,

* @Configure - This class is not only your bootstarp class, but it’s your first and only right now configuration class, java config class. You can come here and specify @Bean that you want and Spring boot will take into consideration over perhaps what it has automatically configured for you, if they are at the same type.
* @ComponentScan: So best practice is not to define your extra beans in here if and when you do that those would be in another classes that you would entity with a configuration
* @Enable auto config.

@Bean - indicates that a method produces a bean to be managed by Spring.

@Service - indicates that an annotated class is a service class.

@Repository - indicates that an annotated class is a repository, which is an abstraction of data access and storage.

@Configuration - indicates that a class is a configuration class that may contain bean definitions.

@Controller - marks the class as web controller, capable of handling the requests.

@RequestMapping - maps HTTP request with a path to a controller method.

@Autowired - marks a constructor, field, or setter method to be autowired by Spring dependency injection.

@SpringBootApplication - enables Spring Boot autoconfiguration and component scanning.

@Component is a generic stereotype for a Spring managed component.

It turns the class into a Spring bean at the auto-scan time.

Classes decorated with this annotation are considered as candidates for auto-detection when using annotation-based configuration and classpath scanning.

@Repository, @Service, and @Controller are specializations of @Component for more specific use cases.

**Difference between @Bean and @Autowired:**

* @Bean tells Spring 'here is an instance of this class, please keep hold of it and give it back to me when I ask'.
* @Autowired says 'please give me an instance of this class, for example, one that I created with an @Bean annotation earlier'.

@ConfigurationProperties(“mfi.ibpm.config”) – This will load all the properties related to “mfi.ibpm.config” from application.properties in classpath.