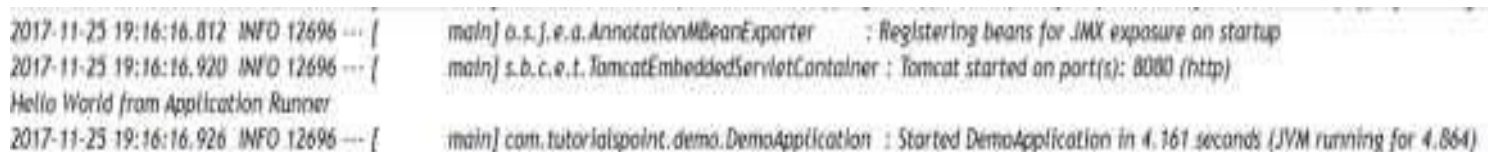


Spring Boot Application Runner:

Application Runner is an interface used to execute the code after the Spring Boot application started.

Sample Code:

```
@SpringBootApplication
public class DemoApplication implements ApplicationRunner {
    public static void main(String[] args) {
        SpringApplication.run(DemoApplication.class, args);
    }
    @Override
    public void run(ApplicationArguments argo) throws Exception {
        System.out.println("Hello World from Application Runner");
    }
}
```



```
2017-11-25 19:16:16.812 INFO 12696 --- [main] o.s.j.e.a.AnnotationMBeanExporter : Registering beans for JMX exposure on startup
2017-11-25 19:16:16.920 INFO 12696 --- [main] s.b.c.e.t.TomcatEmbeddedServletContainer : Tomcat started on port(s): 8080 (http)
Hello World from Application Runner
2017-11-25 19:16:16.926 INFO 12696 --- [main] com.tutorialspoint.demo.DemoApplication : Started DemoApplication in 4.161 seconds (JVM running for 4.864)
```

Spring Boot Command Line Runner:

Command Line Runner is an interface. It is used to execute the code after the Spring Boot application started.

Sample Code:

```
@SpringBootApplication
public class DemoApplication implements CommandLineRunner {
    public static void main(String[] args) {
        SpringApplication.run(DemoApplication.class, args);
    }
    @Override
    public void run(String... argo) throws Exception {
        System.out.println("Hello world from Command Line Runner");
    }
}
```

```
2017-11-25 19:22:33.316 INFO 12460 --- {  
Hello world from Command Line Runner  
2017-11-25 19:22:33.321 INFO 12460 --- {  
main] l.b.c.e.t.TomcatEmbeddedServletContainer : Tomcat started on port(s): 8080 (http)  
main] com.tutorialspoint.demo.DemoApplication : Started DemoApplication in 3.69 seconds (JVM running for 4.083)
```

These runners are used to run the logic on application startup, for example spring boot has `ApplicationRunner`(Functional Interface) with `run` method.

`ApplicationRunner run()` will get execute, just after `applicationcontext` is created and before spring boot application startup.

`ApplicationRunner` takes `ApplicationArgument` which has convenient methods like `getOptionNames()`, `getOptionValues()` and `getSourceArgs()` and `CommandLineRunner` is also a Functional interface with `run` method.

`CommandLineRunner run()` will get execute, just after `applicationcontext` is created and before spring boot application starts up.

It accepts the argument, which are passed at time of server startup.

Both provides the same functionality and the only difference between `CommandLineRunner` and `ApplicationRunner` is `CommandLineRunner.run()` accepts `String array[]` whereas `ApplicationRunner.run()` accepts `ApplicationArguments` as argument.