Actuator

Spring-boot-starter-actuator :: maven dependency

- Actuator uses for monitoring our application, this feature will give like production ready application.
- Add actuator dependency to our pom.xml

```
<dependency>
<groupId>org.springframework.boot</groupId>
<artifactId>spring-boot-starter-actuator</artifactId>
</dependency>
```

- Whenever you will add this dependency, we can get the information(monitor) about our application like

1. We can see the beans list
2. We can see how many calls came to our application
3. Which services call how many times
4. We can see the metrix
5. Health – server is up or not
6. Database details & memory utilization details --------- like this so many we can monitor by using this feature.

We can access through **localhost:8080/actuator**

Note:--

* now we can see the few of the urls for monitoring

* If you add below line in application.properties file, we can see all links to monitor different areas`management.endpoints.web.exposure.include=*`

`Note:--`

```
<dependency>
	<groupId>org.springframework.data</groupId>
	<artifactId>spring-data-rest-hal-browser</artifactId>
</dependency>
```

If you add above dependency we can get the hal browser for monitor our rest urls

NOTE:-- we should not give access to all, we should add below line in application.properties file to restrict access ..

EX: --

/auditevents: we can see the security how many people failed to connect this application.

/beans:: displays all the beans configured in this project..

/health: check the health of our application { status: up }

/metrix: we can get all information here like jvm memory ---

ENDPOINT	USAGE
`/env`	Returns list of properties in current environment
`/health`	Returns application health information.
`/auditevents`	Returns all auto-configuration candidates and the reason why they 'were' or 'were not' applied.
`/beans`	Returns a complete list of all the Spring beans in your application.
`/trace`	Returns trace logs (by default the last 100 HTTP requests).
`/dump`	It performs a thread dump.
`/metrics`	It shows several useful metrics information like JVM memory used, system CPU usage, open files, and much more.

`We may get some performance issues if you add all these features..`

Developer Tools

- If you add devtools dependency to your application in pom.xml
- Whenever you will do any changes in the classpath the server will restart automatically..
- If you stop the server and restart the server will take more time to compare with auto restart