Spring Boot Flow

Flow for Standalone application

- Flow starts from main() method of the SpringBootApplication. From that main method run() method of SpringApplication is called.
- Locates and loads application.properties/yml file
- There application first **check for active Profile**

- So here based on the active profile set in application.properties/yml the Profile related object is created.
- Activate profile related application-<xxx>.properites/.yml file is located and loaded.
- Then based on the application type ApplicationContext object is created.
  If it is standalone→AnnotationConfigApplicationContext (yes)
  If it is web app →AnnotationConfigServletWebServerApplicationContext
- Autoconfiguration based spring bean objs will be created based on the jar files that are added by collecting inputs from the active profile specific properties or yml file

- After AutoConfiguration is done, it creates spring bean object based on configuration class @Bean methods, stereotype annotations having scope singleton by scanning Configuration classes with the support of @Import,@ComponentScan annotations. In this Process the dependency Injections on beans will also be completed.

- All the above beans will be placed Internal Cache/HashMap of IOC container
- All the created objects are registered with JMX as MBeans.Java Management Extensions (**JMX**) is a Java technology that supplies tools for managing and monitoring applications, system objects, devices (such as printers) and service-oriented networks. Those resources are represented by objects called **MBeans** (for Managed Bean).

- Then the remaining logics of main method is getting executed like ctx.getBean(-) and calling b.methods
- closing ApplicationContext container by calling ctx.close();
- All singleton scope bean objects will be destroyed..
- All beans /objects/devices and etc(MBeans) that registred with JMX will be unregistered.