Date : 28/12/2020
Spring Boot 9AM
Mr. RAGHU

Environment :-
System where our application is running with all setup.
Ex: Server , Database, JRE, Application Code..etc

Example Env: Dev, QA, UAT, Prod ..etc

** code + properties

```
--JDBC---
ProdEnv:
Class.forName("MySQLDriver");
Connection con = DriverManager.getConnection("mysql","root","nit");
DevEnv:
Class.forName("OracleDriver");
Connection con = DriverManager.getConnection("orcl","sa","nit");
-------------------------------------------------------------
```

*) If we move our code from One Environment to another Environment,
code (mostly) remains same. But Properties may get changed.

Spring Boot Profiles

Profiles is used to load properties file based on Environment.
ie Define one properties file for one Environment.

```
--Syntax for Properties file---
application-[profileName].yml
application-[profileName].properties
-------------------------------
```

*) If we did not specify any profile, then default profile is active.
default profile properties name is : application.properties

#1 Creating Profile based Properties file

application.properties	---	default profile
application-qa.properties	---	qa profile
application-uat.properties	---	uat profile
application-prod.properties	---	prod profile

#2 Activate Profile while running application

```
spring.profiles.active=<profileName>
```

using command line args, VM args, properties file.

```
--spring.profiles.active=qa        (command line args)
```

***) Need not to modify key=val for different env using single file.
Insted of Modifing existed one, creating new file is better.

***) Profiles is for selecting properties file. To read data use again
@Value or @ConfigurationProperties

---

*) Profiles Fallback : If any key-val is not present in current profile
then same key data will be taken from default profile properties
ie called as Profiles Fallback.

	default	qa	prod
	a=10		a=70
	b=20	b=40	b=80
	c=30	c=60	
Active Prolfile	default	qa	prod
Read and Print	a = 10	a=10	a=70
	b = 20	b=40	b=80
	c = 30	c=60	c=30

*) Fallback is advantage only. Consider default profile is having 100 keys
in qa we need only 10 keys with different value, reamining 90 are same
as default, then define qa properties with 10 keys only.

--------------------------code--------------------------------------

```
#1. Create Project
Name : SpringBoot2ProfileFirstApp
Dep  : Lombok

#2. Runner class
package in.nareshit.raghu.runner;

import org.springframework.beans.factory.annotation.Value;
import org.springframework.boot.CommandLineRunner;
import org.springframework.stereotype.Component;

import lombok.ToString;

//ctrl+shift+O
@Component
@ToString
public class ProfileDataRunner implements CommandLineRunner {
        @Value("${my.app.title}")
        private String title;

        @Value("${my.app.loc}")
        private String loc;
        @Override
        public void run(String... args) throws Exception {
                System.out.println(this);
        }


}
```

S#3

```
--application.properties--
my.app.title=NIT-DEFAULT
my.app.loc=HYD-DEFAULT

--application-qa.properties--
#my.app.title=NIT-QA
my.app.loc=CHN-QA

--application-prod.properties--
my.app.title=NIT-PROD
my.app.loc=HYD-PROD
```