How to Read value From application.properties in spring boot

lHere we see how we can **read application.properties in spring boot .** We have different ways for reading values from application properties ,here we will discuss three ways for reading keys from application.propeties file as following

- How to read values from application.properties Using **@Value** annotation
- How to read values from application.properties Using @**ConfigurationProperties** annotation
- How to read values from application.properties Using Environment object

How to read value from application.properties Using @Value

Now we see how to **read properties file in spring using annotation** @Value . If you are in new spring boot and then i will recommencement to you first

read how create project in Spring boot . Firstly we will add below values in application.properties file of our project.

Application.properties

```
email.username=javavogue
email.pwd=12345
```

Now we will create java Class with name EmailService where we map these key to class's field using @Value as below .

```
package com.javavogue.demo.service;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.beans.factory.annotation.Value;
import org.springframework.stereotype.Service;

@Service
public class EmailService {

    @Value("${email.username}")
    private String username;

    @Value("${email.pwd}")
    private String pwd;

    @Autowired
    private EmailConfig emailConfig;

   public void sendEmail(){
```

```
    System.out.println("reading value from propertes file using @value annotation");
    System.out.println("username ="+ username);
    System.out.println("pwd ="+ pwd);

    }

}
```

How to read value from application.properties Using @ConfigurationProperties

In this we explain **How to load properties using Spring Boot @ConfigurationProperties** .In this way we will create a plain java object where each class field name same as the key of application.properties. Because In application.properties file we have email string as prefix of key so that we will use email string with @ConfigurationProperties annotation . In below we have given **Spring Boot @ConfigurationProperties example**

EmailConfig.java

```
       package com.javavogue.demo.service;

import javax.validation.constraints.Max;
import javax.validation.constraints.Min;

import org.springframework.boot.context.properties.ConfigurationProperties;
import org.springframework.stereotype.Component;

@Component
```

```
@ConfigurationProperties("email")
public class EmailConfig {

  private String   username;

    private String  pwd;

    public String getUsername() {
        return username;
    }
    public void setUsername(String username) {
        this.username = username;
    }
    public String getPwd() {
        return pwd;
    }
    public void setPwd(String pwd) {
        this.pwd = pwd;
    }

    @Override
    public String toString() {
        return "EmailConfig [username=" + username + ", pwd=" + pwd + "]";
    }

}
```

How to read application.properties Using using Environment object

Now we will read value from application.propeties file using Environment object. We will pass key in getProperty("Key") method on environment

object as

```
      package com.javavogue.demo.controller;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.core.env.Environment;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.RestController;

import com.javavogue.demo.service.EmailService;

@RestController
public class EmailController {

    @Autowired
    private Environment env;

    @GetMapping("/sendmail")
    public void sendMail(){
System.out.println("values from application properties using Environment ");
System.out.println("username ="+ env.getProperty("email.username"));
System.out.println("pwd ="+ env.getProperty("email.pwd"));
    }
}
```