

Java Configuration Class

Let's see the steps to create java configuration file

Step 1: Create a class CarConfig

```
public class CarConfig {  
  
}
```

Step 2: To specify the class is a configuration file you need to mention that using **@Configuration** as shown below

```
@Configuration  
public class CarConfig {  
  
}
```

Step 3: Now to scan all the beans present inside the package you need to make use of **@ComponentScan**("Mention the path of package from where to scan bean") as shown below

```
@Configuration  
@ComponentScan("com.tapacad.spring")  
public class CarConfig {  
  
}
```

Step 4: To include property file in the java configuration we need to make use of **@PropertySource**("classpath:car.properties") as shown below

```
@Configuration  
@ComponentScan("com.tapacad.spring")  
@PropertySource("classpath:car.properties")  
public class CarConfig {  
  
}
```

Step 5: Now to load the java configuration class we need to create an object of **AnnotationConfigApplicationContext** instead of **ClassPathXmlApplicationContext** to load Application Context

```
AnnotationConfigApplicationContext context = new  
AnnotationConfigApplicationContext(CarConfig.class);
```

Step 6: Now let's create main method and try to create bean and print the bean

MyApp10.java

```
package com.tapacad.spring;  
  
import  
org.springframework.context.annotation.AnnotationConfigApplicationContext;  
t;  
  
public class MyApp10 {  
  
    public static void main(String[] args) {  
        //      Load CarConfig configuration file  
        AnnotationConfigApplicationContext context =  
            new  
AnnotationConfigApplicationContext(CarConfig.class);  
  
        //      Get bean  
        Car car = context.getBean("bmw", Car.class);  
  
        //      print car  
        System.out.println(car);  
  
        //      close context  
        context.close();  
  
    }  
}
```

```
}
```

Output:

```
Bean Created  
com.tapacad.spring.BMW@78ffe6dc  
Bean Destroyed
```

As you can see from the above output we have successfully created the bean and destroyed using **java Configuration file**