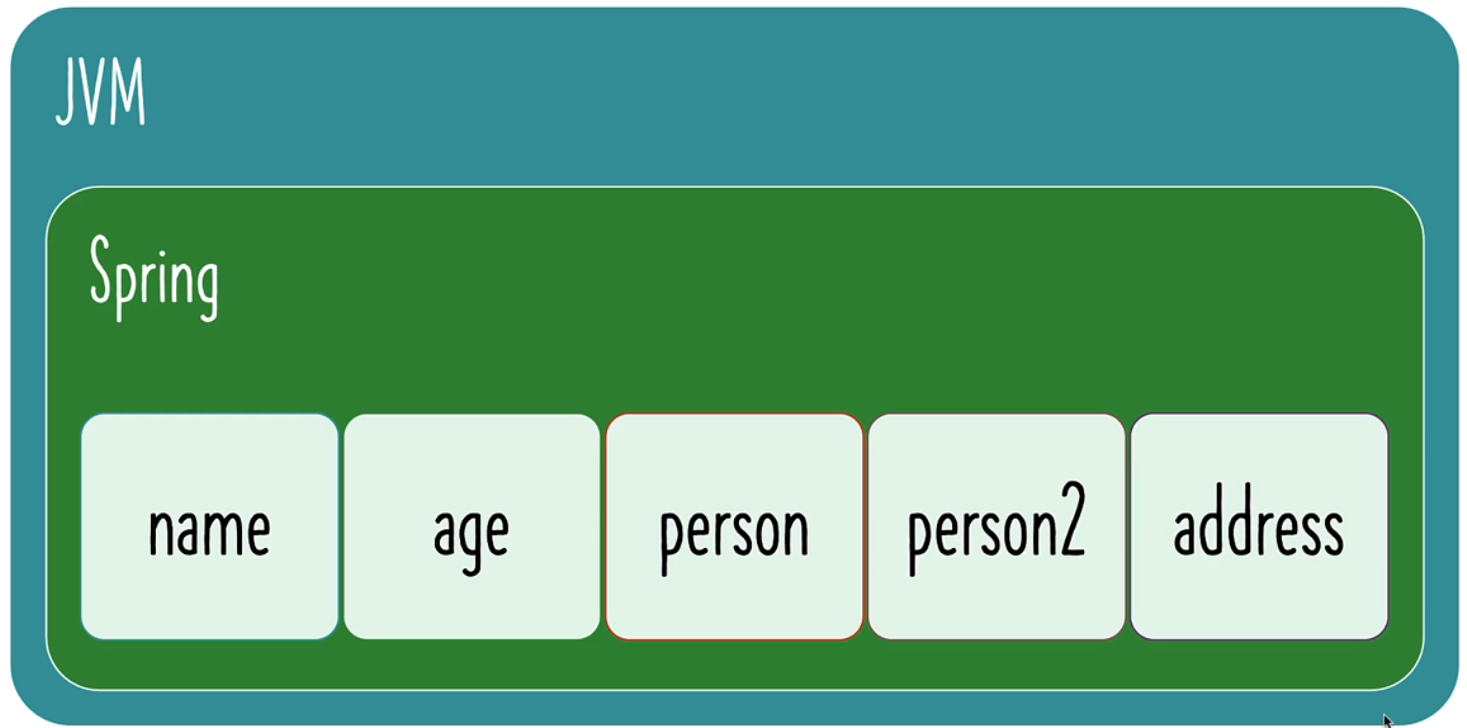
**YOUR FIRST JAVA SPRING BEAN AND LAUNCHING JAVA SPRING CONFIGURATION**

We can begin with a simpler example by having **Spring framework** manage basic objects for us.



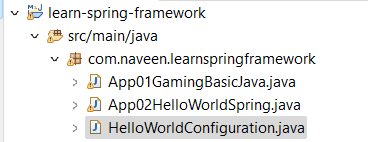
Our objective is to **create a Spring bean** named "**name**", with a string value, and have the **Spring framework manage it** for us **within the JVM** by setting up the **Spring context**.



**How can we do this?**

1. **Launch a Spring Context.**
   1. By using the class **AnnotationConfigApplicationContext**: With this we can create the spring context.
   2. So, inside the JVM we want to create a spring context. And we are using a Configuration class to create a AnnotationConfigApplicationContext.
   3. So, create a new instance for this class AnnotationConfigApplicationContext using our Configuration class.
2. **Configure the things that we want spring to manage.** 
   1. One of the approaches that we can use to configure the things that spring needs to manage is called **Configuration class**.
   2. So, we can create a Configuration class and have everything defined.
   3. For example, “name” defined in this class, and we can use the configuration class to launch the spring context.

**Project Structure:**



**HelloWorldConfiguration.java**

package com.naveen.learnspringframework;

import org.springframework.context.annotation.Configuration;

@Configuration

public class HelloWorldConfiguration {

}

Now, **HelloWorldConfiguration** class is a spring configuration class. Using **@Configuration** We can indicate the class as **spring configuration class**. In this class, we can define spring beans.

**Spring beans**: The things which are managed by spring is spring beans.

We can define the **methods to create the spring beans**, in our **Configuration** class.

**App02HelloWorldSpring.java**

package com.naveen.learnspringframework;

import org.springframework.context.annotation.AnnotationConfigApplicationContext;

public class App02HelloWorldSpring {

    public static void main(String[] args) {

        // Launch a Spring Context.

        var context =

            new AnnotationConfigApplicationContext(HelloWorldConfiguration.class);

    }

}

Now we are in this stage,

We have **JVM** and launch up the **spring context**.

What we want to tell spring is **to manage a bean**, a “name” object. How can we do that?

Treemap chart

Description automatically generated with medium confidence

**HelloWorldConfiguration.java**

package com.naveen.learnspringframework;

import org.springframework.context.annotation.Bean;

import org.springframework.context.annotation.Configuration;

@Configuration

public class HelloWorldConfiguration {

    @Bean

    //Indicates that a method produces a bean to be managed by the Spring container.

    public String name() {

        return "Naveen";

    }

}

We have created the **name** object and it is managed by **spring**. We can ask the **spring** **context** to give the **name** object. Anything that is **managed by spring** is called a **bean**. **getBean ()** method used to **retrieving Beans** managed by Spring. There are multiple ways to retrieve values from the context. One of the ways is give **method** name within double quotes.

**App02HelloWorldSpring.java**

package com.naveen.learnspringframework;

import org.springframework.context.annotation.AnnotationConfigApplicationContext;

public class App02HelloWorldSpring {

    public static void main(String[] args) {

        // Launch a Spring Context.

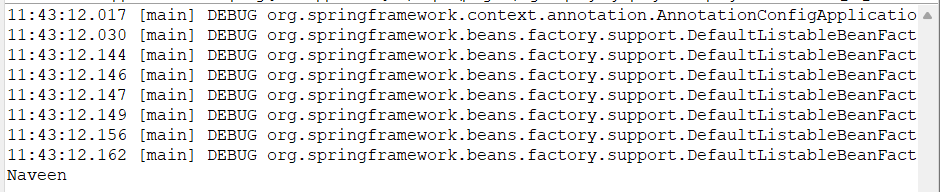
        var context =

            new AnnotationConfigApplicationContext(HelloWorldConfiguration.class);

        System.out.println(context.getBean("name"));

}

**OUTPUT:**

****