

## JavaBeans Example

Consider a car class with a few characteristics:

```
package com.myCar;

public class CarsBean implements java.io.Serializable {
    private String carName = null;
    private String brandName = null;
    private int price = 0;

    public CarsBean() {
    }
    public String getCarName(){
        return carName;
    }
    public String getBrandName(){
        return brandName;
    }
    public int getPrice(){
        return price;
    }
    public void setCarName(String carName){
        this.carName = carName;
    }
    public void setBrandName(String brandName){
        this.brandName = brandName;
    }
    public void setPrice(Integer price){
        this.price = price;
    }
}
```

## How do we access JavaBeans in JSP?

The useBean action creates a JavaBean object that may be used in a JSP. The bean becomes a scripting variable once it is declared, and it may be accessed by both scripting elements and other custom tags in the JSP. The useBean tag has the following full syntax:

```
<jsp:useBean id = "bean's name" scope = "bean's scope" typeSpec/>
```

Depending on our needs, the scope attribute can be set to a page, request, session, or application. The id attribute's value can be anything as long as it's a distinct name among other useBean declarations in the same JSP.

The useBean action is demonstrated in the example below:

```
<html>
<head>
    <title>JavaBeans Sample Example</title>
</head>

<body>
    <jsp:useBean id = "date" class = "java.util.Date" />
    <p>The date/time is <%= date %>
</body>
</html>
```

We will get the following outcome:

**The date/time is Mon Jan 30 09:20:44 GST 2022**

## Getting to the Properties of JavaBeans

We can utilise the `<jsp:getProperty/>` action to access the get methods and the `<jsp:setProperty/>` action to access the set methods in addition to the `<jsp:useBean...>` action. Here is the complete syntax:

```
<jsp:useBean id="idHere" class="beansClass" scope="scopeOfBeans">
  <jsp:setProperty name="beansId" property="nameOfTheProperty" value="valueHere" />
  <jsp:getProperty name="beansId" property="nameOfTheProperty" />
  .....
</jsp:useBean>
```

The id of a JavaBean previously introduced to the JSP by the useBean operation is referenced by the name attribute. The name of the get or set methods that should be used is specified in the property attribute.

The following example demonstrates how to use the above syntax to obtain data:

```
<html>
<head>
  <title>Example of getting and setting properties</title>
</head>

<body>
  <jsp:useBean id="cars" class="com.myCar.CarsBean">
    <jsp:setProperty name="cars" property="carName" value="Swift" />
    <jsp:setProperty name="cars" property="brandName" value="Maruti Suzuki" />
    <jsp:setProperty name="cars" property="price" value="700000" />
  </jsp:useBean>

  <p>Car Name:
    <jsp:getProperty name="cars" property="CarName" />
  </p>

  <p>Brand Name:
    <jsp:getProperty name="cars" property="brandName" />
  </p>

  <p>Price:
    <jsp:getProperty name="cars" property="price" />
  </p>

</body>

</html>
```

Let's add CarsBean.class to the CLASSPATH variable. The JSP mentioned above can be accessed. The following outcome will be shown:

**Car Name: Swift**

**Brand Name: Maruti Suzuki**

**Price: 700000**