



ROTEIRO DE LABORATÓRIO - BEANS

JavaBeans são componentes de *software* escritos na linguagem de programação Java. Segundo a especificação da Sun Microsystems os JavaBeans são "componentes reutilizáveis de *software* que podem ser manipulados visualmente com a ajuda de uma ferramenta de desenvolvimento". Um bean também pode ser definido como uma classe Java que expõe propriedades, seguindo uma convenção de nomenclatura simples para os métodos getter e setter. Praticamente são classes escritas de acordo com uma convenção em particular. São usados para encapsular muitos objetos em um único objeto (o bean), assim eles podem ser transmitidos como um único objeto em vez de vários objetos individuais. O JavaBean é um Objeto Java que é serializável, possui um construtor nulo (construtor vazio - construtor noarg) e permite acesso às suas propriedades através de métodos getter e setter.

COMO PREENCHER AUTOMATICAMENTE OBJETOS JAVA A PARTIR DE PARÂMETROS DE REQUISIÇÃO: BEANS DE FORMULÁRIO

O método `getParameter` facilita a leitura de parâmetros de requisição de chegada. Uma vez que o valor de retorno de `getParameter` é uma `String`, você terá que analisar sintaticamente o valor você mesmo se quiser outros tipos de valores. Além disso, o uso desse método é muito difícil quando se lida com grande número de parâmetros.

Felizmente o pacote Jakarta Commons contém classes que facilitam a construção de um utilitário para associar automaticamente parâmetros de uma requisição com propriedades dos beans.

Para tanto, faça o passo a passo a seguir.

1) adicione a dependência seguinte no seu arquivo `pom.xml`

```
<dependency>
  <groupId>commons-beanutils</groupId>
  <artifactId>commons-beanutils</artifactId>
  <version>1.9.4</version>
</dependency>
```

2) Crie a classe `BeanUtilities`

```
import java.util.Map;
import javax.servlet.http.HttpServletRequest;
import org.apache.commons.beanutils.BeanUtils;
public class BeanUtilities {

    public static void populateBean(Object formBean, HttpServletRequest request) {
        populateBean(formBean, request.getParameterMap());
    }
}
```

```

    }

    public static void populateBean(Object bean, Map propertyMap) {
        try {
            BeanUtils.populate(bean,propertyMap);
        }catch (Exception e)
        {

        }

    }

}

```

3) Use o método populate da classe BeanUtilities para atribuir os parâmetros de requisição ao bean carro.

```

protected void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
    // TODO Auto-generated method stub
    //String atributo = request.getParameter("nome");
    PrintWriter out = response.getWriter();

    Carro c = new Carro();
    BeanUtilities.populateBean(c, request);
    out.println("<html>");

    out.println("<h1>" + c.getAno() + "</h1>");
    out.println("<h1>" + c.getMarca() + "</h1>");
    out.println("<h1>" + c.getModelo() + "</h1>");

    out.println("</html>");

}

```