

4.5. Reescrita de URL

Nem todos os navegadores suportam cookies ou podem estar configurados para bloqueá-los. Para contornar essa limitação e ainda manter o controle de sessão (via `HttpSession`), o **servidor pode empregar a técnica de reescrita de URL**.


Essa técnica consiste em **inserir a ID da sessão diretamente na URL** das requisições feitas pelo cliente. Quando o `HttpSession` é criado, ele recebe um identificador único (JSESSIONID). Se os cookies estiverem desativados, esse ID pode ser incluído na URL de cada requisição.

Exemplo prático:

```
out.println("<form action=\"" + response.encodeURL("/JSPPage") + "\">");
```

Este comando **codifica a URL** inserindo a ID da sessão automaticamente se os cookies estiverem desabilitados, resultando em uma URL parecida com esta:

```
http://localhost:8080/exercicio02/JSPPage;jsessionid=30F76DE281982B57B4BEFC2434B6F5E
```

 O método `encodeURL()` verifica se os cookies estão ativados. Se não estiverem, ele automaticamente adiciona o `jsessionId` à URL.

4.6. Escopos de Armazenamento de Dados no Servidor

No ambiente de Servlets e JSPs, os dados podem ser armazenados em diferentes **escopos (áreas de vida e visibilidade)**. São eles:

Escopo	Objeto	Tempo de vida	Visível por quem?
Request	<code>HttpServletRequest</code>	Durante a requisição (ex: do formulário até a resposta)	Apenas pelo servlet ou JSP que está tratando a requisição atual
Session	<code>HttpSession</code>	Enquanto durar a sessão do usuário (ou até expirar)	Por qualquer servlet ou JSP da mesma sessão
Application	<code>ServletContext</code>	Enquanto o servidor estiver em execução	Por todos os componentes da aplicação web (global)

A **Figura 36** mencionada (não exibida aqui) representa graficamente a relação entre os objetos `Request`, `Session` e `Context` dentro de um Web Container.

