

Jakarta EE Web Total


Conhecendo os Listeners




Softblue


cursos online




Tópicos Abordados

- O que são listeners
- Tipos de listeners
- Configurando um listener
- Detalhamento de cada tipo de listener



O que são Listeners

- Os listeners permitem que o seu código seja avisado quando algum evento do seu interesse ocorre
- Se você criar e registrar um listener, o container o invoca na presença de um evento



Tipos de Listeners

- Existem oito tipos de listeners

Interfaces dos Listeners

- ServletContextListener
- ServletContextAttributeListener
- ServletRequestListener
- ServletRequestAttributeListener
- HttpSessionListener
- HttpSessionAttributeListener
- HttpSessionBindingListener
- HttpSessionActivationListener

Os listeners são interfaces. Basta implementar a interface desejada e registrá-la

Configurando um Listener

- Para que o container saiba da existência do listener, é preciso registrá-lo no *web.xml*

```

<web-app>
  <listener>
    <listener-class>listener.MyListener</listener-class>
  </listener>
</web-app>
  
```

Esta classe deve implementar uma das sete interfaces de listener

Listeners e Annotations

- Outra forma de configurar um listener é através de annotations

@WebListener define que a classe é um listener

```

MyListener
@WebListener
public class MyListener implements ServletContextListener {
    public void contextInitialized(ServletContextEvent e) {
    }
    public void contextDestroyed(ServletContextEvent e) {
    }
}
  
```

Listeners do Servlet Context



- **ServletContextListener**

Método	Invocado quando...
contextInitialized()	A aplicação web está sendo iniciada
contextDestroyed()	A aplicação web está sendo terminada

- **ServletContextAttributeListener**

Método	Invocado quando...
attributeAdded()	Um atributo é adicionado ao servlet context
attributeRemoved()	Um atributo é removido do servlet context
attributeReplaced()	Um atributo é substituído no servlet context



Listeners do Servlet Request



- **ServletRequestListener**

Método	Invocado quando...
requestInitialized()	A request está iniciando
requestDestroyed()	A request está terminando

- **ServletRequestAttributeListener**

Método	Invocado quando...
attributeAdded()	Um atributo é adicionado à request
attributeRemoved()	Um atributo é removido da request
attributeReplaced()	Um atributo é substituído na request



Listeners da HTTP Session



- **HttpSessionListener**

Método	Invocado quando...
sessionCreated()	Uma session está sendo criada
sessionDestroyed()	Uma session está sendo invalidada

- **HttpSessionAttributeListener**

Método	Invocado quando...
attributeAdded()	Um atributo é adicionado à session
attributeRemoved()	Um atributo é removido da session
attributeReplaced()	Um atributo é substituído na session



Listeners da HTTP Session



• HttpSessionActivationListener

Método	Invocado quando...
<code>sessionDidActivate()</code>	Uma session está sendo trazida para a memória
<code>sessionWillPassivate()</code>	Uma session está sendo retirada da memória

- Servidores de aplicações podem decidir o que fazer com dados da session
 - Remover da memória (armazenar em disco)
 - Migrar os dados para outra JVM (ambiente distribuído)



Listeners da HTTP Session



• HttpSessionBindingListener

- É o único listener que não precisa ser configurado
 - Nem no `web.xml` nem via `@WebListener`
- Classes cujos objetos precisam ser notificados quando são adicionados ou removidos de uma session implementam esta interface

Método	Invocado quando...
<code>valueBound()</code>	O objeto é adicionado numa session
<code>valueUnbound()</code>	O objeto é removido de uma session

