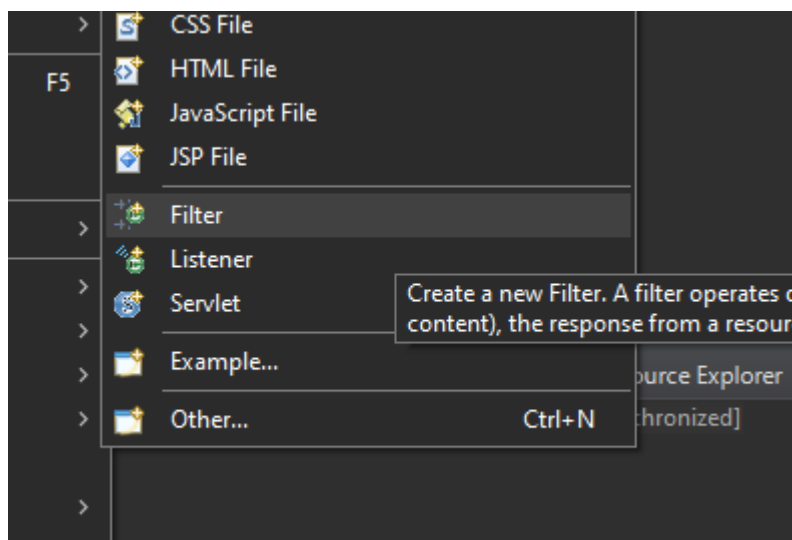


Filtros y Listeners

Implementación Filtro

Primero crearemos un nuevo proyecto dinámico de Java y tras eso crearemos el index.html (Se da por sabido) Tras eso iremos a JavaResources y crearemos el filtro (recordar cambiar la URL a todo minúsculas dándole en el botón Next y editar):

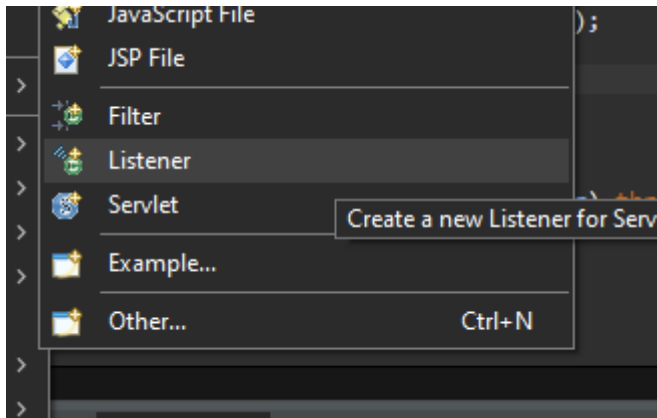


Una vez creado el filtro debemos buscar la siguiente sección para crear nuestro filtro:

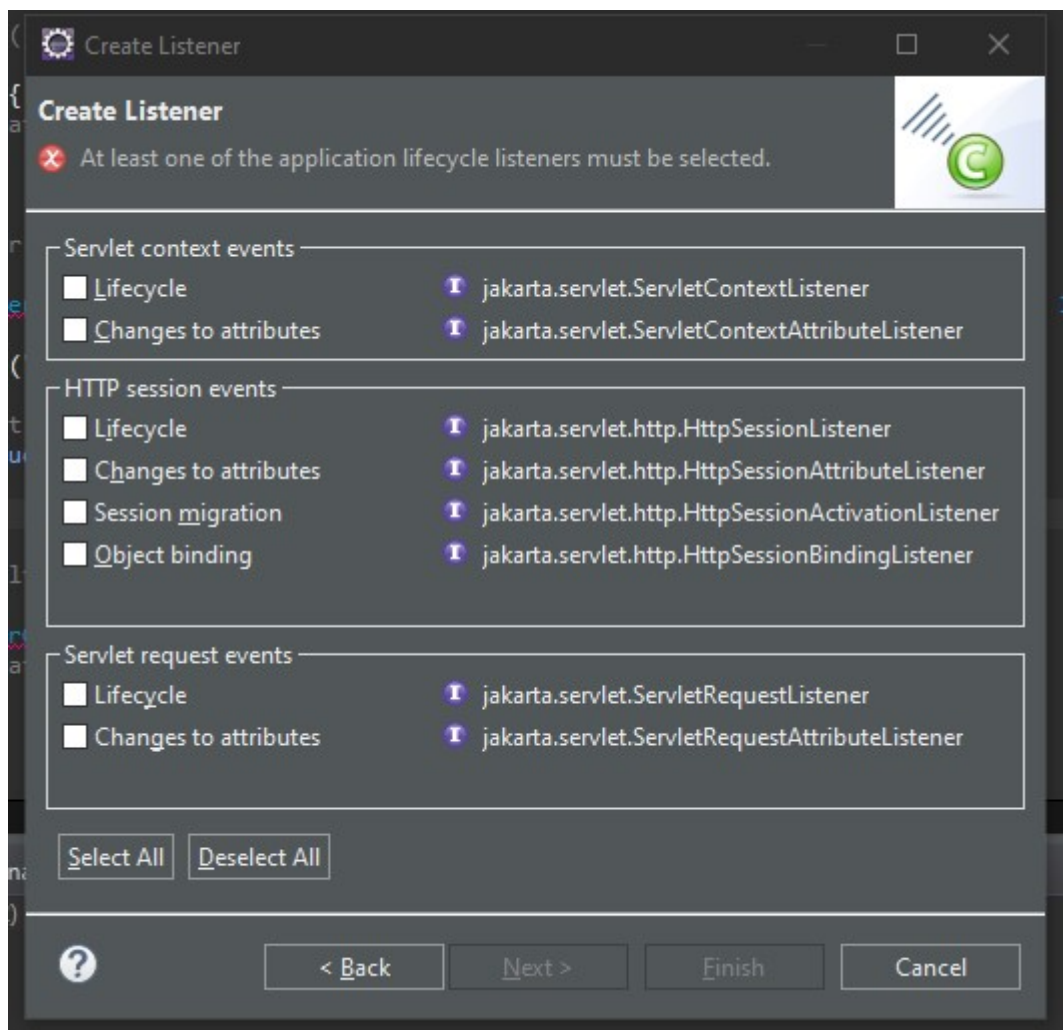
```
public void doFilter(ServletRequest request, ServletResponse response, FilterChain chain) throws IOException, ServletException {  
    System.out.println("Ejecutar filtrar");  
    // pass the request along the filter chain  
    chain.doFilter(request, response);  
}
```

Implementación Listener

Usando el mismo proyecto que antes como base iremos a Java resources y crearemos el listener:



Tras establecer los paquetes y el nombre le daremos a next, donde veremos los siguiente:



Aquí estableceremos que escucha el listener.

ejemplo en Java (notación)

```
//tipo Forward
```

```
@WebFilter(dispatchTypes = {DispatcherType.FORWARD}, urlPatterns = {"/*"})
```

Ejemplo en web.xml

```
<filter>  
<filter-name>Mi filtro</filter-name>  
<filter-class></filter-class>  
  
</filter>
```

Listener

- Se utilizan para escuchar eventos que se producen durante la ejecución de la aplicación.
- Se pueden configurar con notación o en el Web.xml

```
@WebListener  
public class MiListener implements ServletContextListener, AsyncListener;
```

From:
<http://www.knoppia.net/> - **Knoppia**

Permanent link:
http://www.knoppia.net/doku.php?id=dad2:filtros_y_listeners

Last update: **2024/02/27 11:38**

