

JAVA.UTIL.SERVICELOADER.TOSTRING METHOD

http://www.tutorialspoint.com/java/util/serviceloader_tostring.htm

Copyright © tutorialspoint.com

Description

The **java.util.ServiceLoader.toString** method returns a string describing this service.

Declaration

Following is the declaration for **java.util.ServiceLoader.toString** method

```
public String toString()
```

Parameters

- NA

Return Value

This method returns a descriptive string

Exception

- NA

Example

In order the service to be registered, we need a META-INF/service folder in our classpath. In this particular folder, we need a text file with the name of the interface we implementing containing a single line listing the concrete class name of the implementation. In our case the name of the file is **com.tutorialspoint.ServiceProvider** and contains this line:

```
com.tutorialspoint.ServiceImplementation
```

Our service code is the following:

```
package com.tutorialspoint;

public class ServiceImplementation extends ServiceProvider {

    public String getMessage() {
        return "Hello World";
    }
}
```

The following code loads the service that is registered and uses it to get the message from the service:

```
package com.tutorialspoint;

import java.util.ServiceLoader;

public abstract class ServiceProvider {

    public static ServiceProvider getDefault() {

        // load our plugin
        ServiceLoader serviceLoader =
            ServiceLoader.load(ServiceProvider.class);

        // print the serviceLoader as a string
        System.out.println(""+serviceLoader.toString());
    }
}
```

```

// checking if load was successful
for (ServiceProvider provider : serviceLoader) {
    return provider;
}
throw new Error("Something is wrong with registering the addon");
}

public abstract String getMessage();

public static void main(String[] ignored) {
    // create a new provider and call getMessage()
    ServiceProvider provider = ServiceProvider.getDefault();
    System.out.println(provider.getMessage());
}
}

```

Let us compile and run the above program, this will produce the following result:

```

java.util.ServiceLoader[com.tutorialspoint.ServiceProvider]
Hello World

```

```

Loading [MathJax]/jax/output/HTML-CSS/fonts/TeX/fontdata.js

```