Java EE:

Servlets API 4.0

SoftUni Team

Technical Trainers

Software University

http://softuni.bg

Table of Contents

Questions

Software University

sli.do

#java-web

Java Servlets

Overview

What are Servlets?(1)

- Servlets:
  - Component-based
  - Platform-independent
  - Method for building Web based applications
  - Have access to the entire family of Java APIs

What are Servlets?(2)

- Java Servlets:
  - Programs that run on a Web or Application server
  - Middle layer between a requests
- Servlets can:
  - Collect input
  - Present records from a database or another source
  - Create web pages dynamically

Servlets Tasks(1)

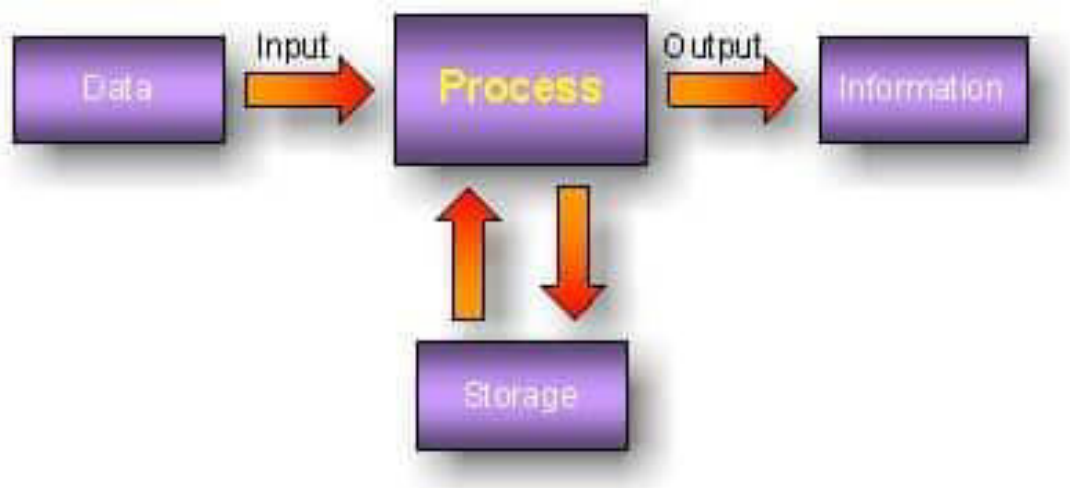
- Read the explicit data sent by the clients (browsers):
  - HTML form on a Web page
  - Applet
  - Custom HTTP client program

Servlets Tasks(2)

- Read the implicit HTTP request data sent by the clients (browsers):
  - Cookies
  - Media types
  - Compression schemes the browser understands

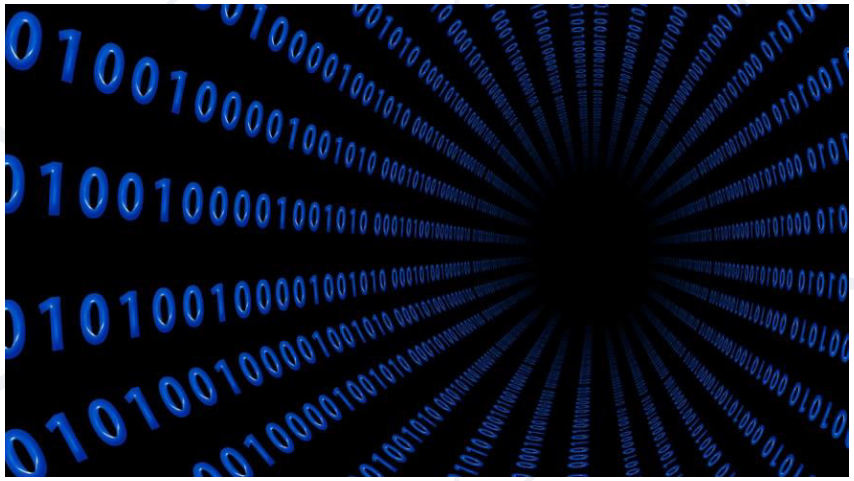
Servlets Tasks(3)

- Process the data and generate the results:
  - Talking to a database
  - Invoking a Web service
  - Computing the response directly



Servlets Tasks(4)

- Send the explicit data to the clients (browsers):
  - Text (HTML or XML)
  - Binary (GIF images)
  - Excel



Servlets Tasks(5)

- Send the implicit HTTP response to the clients (browsers):
  - Telling the browsers or other clients what type of document is being returned (e.g., HTML)
  - Setting cookies
  - Caching parameters

Servlet Life Cycle

- A servlet life cycle:
  - init() method – initialize servlet
  - service() method - process client's request
  - destroy() method - servlet is terminated

The init() Method

- The init() method:
  - Called only once when the servlet is created
- Servlet is created when a user first invokes a URL corresponding to the servlet

The service() Method

- The service() method:
  - The main method to perform the actual task
  - Called by the servlet container to:
    - Handle requests coming from the client
    - Write response back to the client
  - Checks the HTTP request type (GET, POST, PUT, DELETE, etc.) and calls the appropriate method – doGet(), doPost(), etc.

The doGet() Method

- A GET request:
  - Results from:
    - Normal request for a URL
    - HTML form that has no METHOD specified
  - Should be handled by doGet() method

Servlet.java

```
@WebServlet("/")
public class Servlet extends HttpServlet {
  protected void doGet(…) {
    // Servlet code
  }
}
```

The doPost() Method

- A POST request:
  - Results from an HTML form
  - Should be handled by doPost() method.

Servlet.java

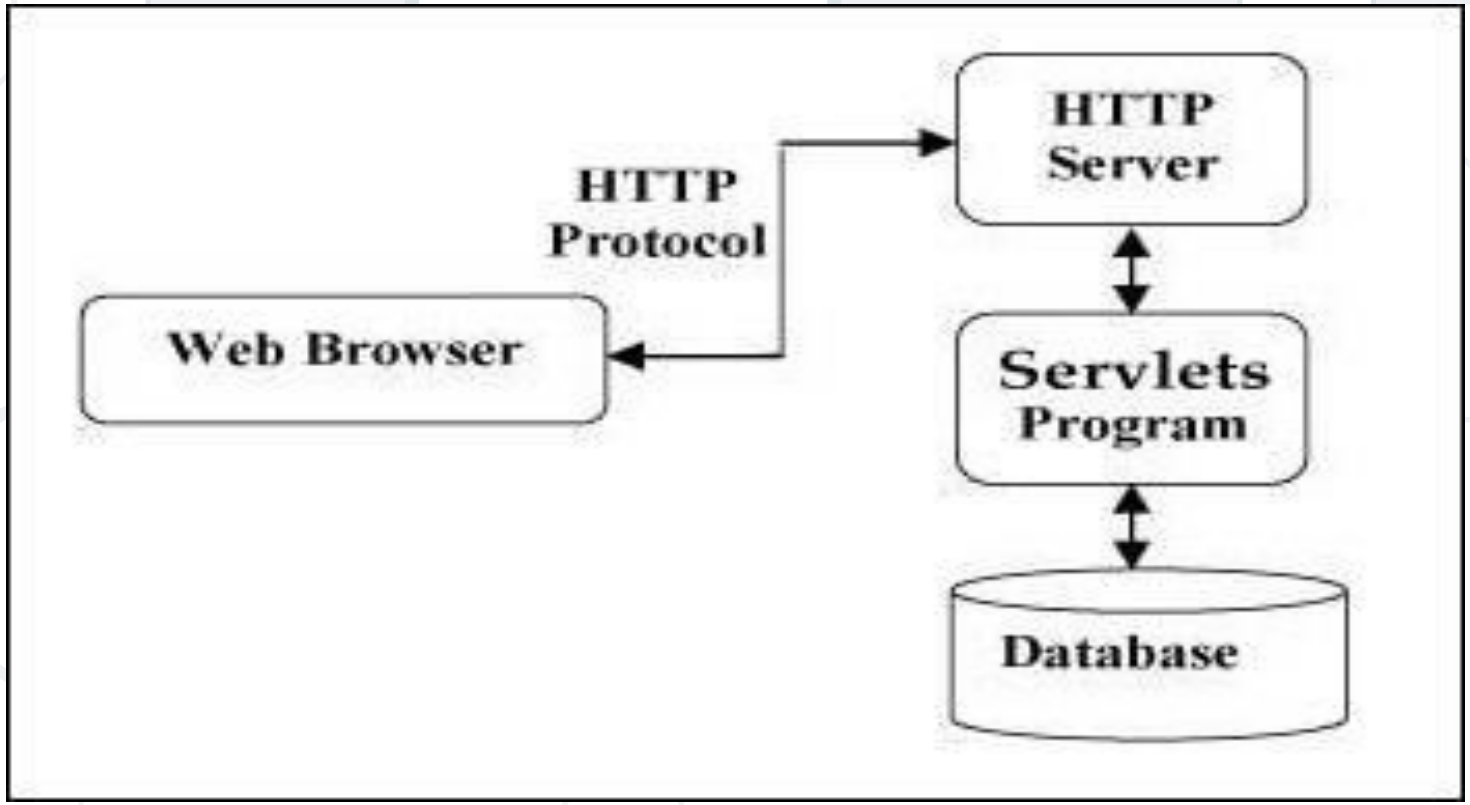
```
@WebServlet("/")
public class Servlet extends HttpServlet {
  protected void doPost(…) {
    // Servlet code
  }
}
```

The destroy() Method

- The destroy() method:
  - Called only once at the end of the lifecycle of a servlet
  - Gives your servlet chance to:
    - Close database connections
    - Halt background threads
    - Write cookie lists
    - Hit counts to disk
    - Perform other such cleanup activities

Servlets Architecture

- The following diagram shows the position of Servlets in a Web Application:



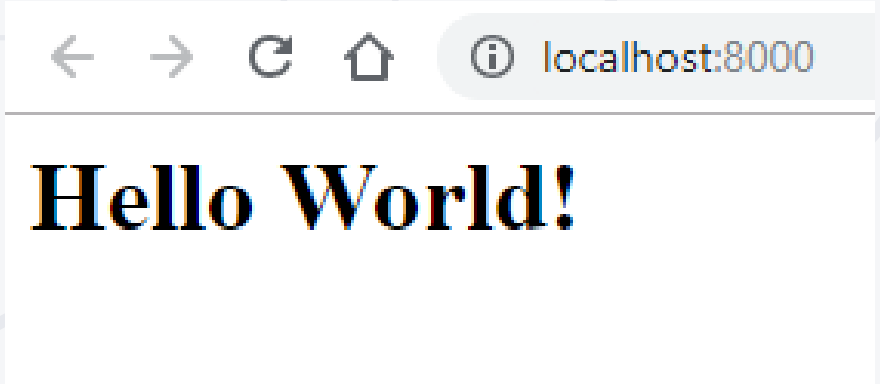
Java Servlets

Examples

Hello World!

Servlet.java

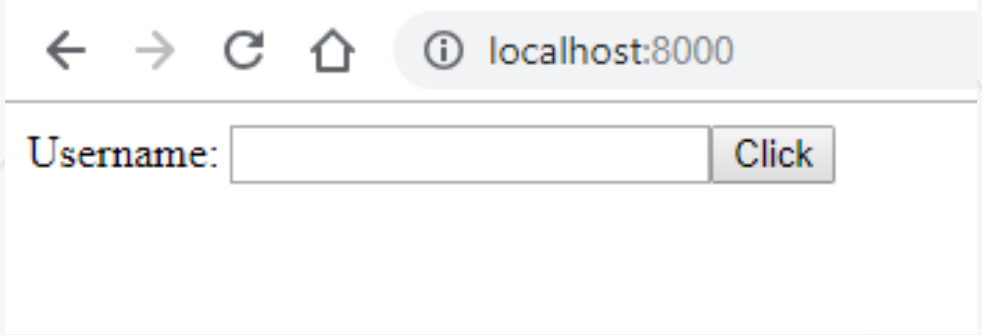
```
@WebServlet("/")
public class Servlet extends HttpServlet {

  private String message;

  protected void init(…) {
    this.message = “Hello World!”;
  }

  protected void doGet(…) {
    response.setContentType(“text/html”);
    PrintWriter out = response.getWriter();
    out.println(String.format(“<h1>%s</h1>”, this.message));
  }
}
```



Greeting!(1)

FormServlet.java

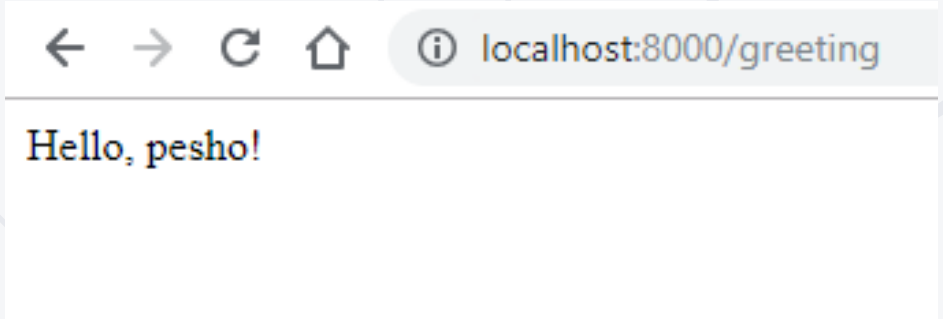
```
@WebServlet("/")
public class FormServlet extends HttpServlet {
  private String form;
  protected void init(…) {
    this.form = "" +
                "<form action=\"/greeting\" method=\"post\">" +
                    "<label>Username: " +
                    "<input name=\"username\"/></label>" +
                    "<button>Click</button>" +
                "</form>";
  }
  protected void doGet(…) {
    response.setContentType("text/html");
    PrintWriter out = response.getWriter();
    out.println(this.form);
  }
}
```



Greeting!(2)

GreetingServlet.java

```
@WebServlet("/greeting")
public class GreetingServlet extends HttpServlet {
  protected void doGet(…) {
    response.setContentType("text/html");
    response.setStatus(200);
 }

  protected void doPost(…) {
    response.setContentType("text/html");

    PrintWriter out = response.getWriter();
    out.println(String.format("Hello, %s!", request.getParameter("username")));
  }
}
```



Hibernate

Overview

What is JDBC?

- JDBC:
  - Java Database Connectivity
  - Provides a set of Java API for accessing the relational databases from Java program
  - Enables Java programs to:
    - Execute SQL statements
    - Interact with any SQL compliant database

What is Hibernate?(1)

- Hibernate:
  - **O**bject-**R**elational **M**apping (ORM) solution for Java
  - Powerful
  - High performance
  - Object-Relational Persistence
  - Query service
  - Maps:
    - Java classes to database tables
    - Java data types to SQL data types

What is Hibernate?(2)

- Hibernate:
  - Sits between traditional Java objects and database server
  - Handles all the works in persisting objects based on the appropriate O/R mechanisms and patterns

Java EE and Hibernate(1)

- Hibernate dependency:

pom.xml

```
<dependency>
    <groupId>org.hibernate</groupId>
    <artifactId>hibernate-core</artifactId>
    <version>5.4.0.Final</version>
</dependency>
```

- Hibernate .jar file: http://central.maven.org/maven2/org/hibernate/hibernate-core/5.4.0.Final/hibernate-core-5.4.0.Final.jar
  - Put the .jar file into TomEE/lib folder.

Java EE and Hibernate(2)

Software University

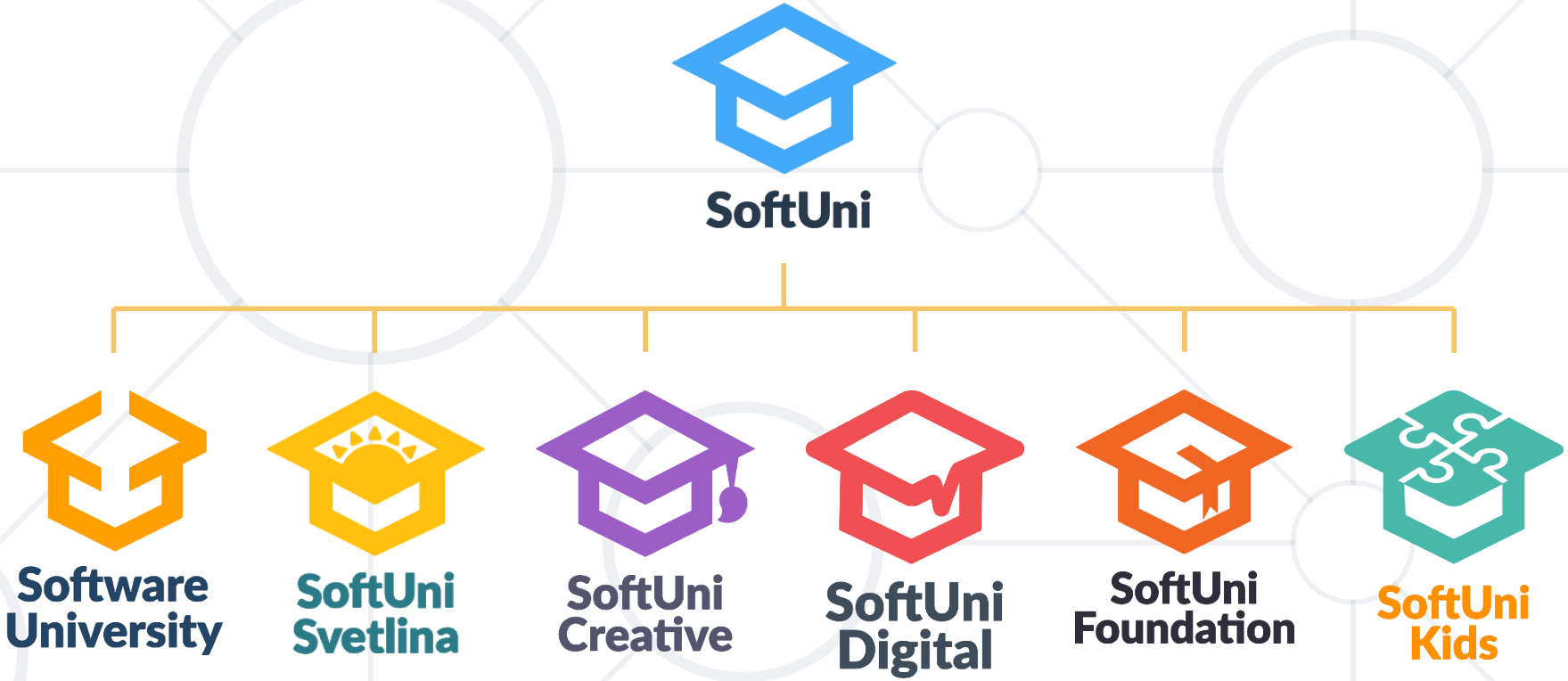
persistence.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<persistence xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xmlns="http://xmlns.jcp.org/xml/ns/persistence"
          xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/persistence
          http://xmlns.jcp.org/xml/ns/persistence/persistence_2_1.xsd" version="2.1">
          <persistence-unit name="soft_uni" transaction-type="RESOURCE_LOCAL">
                    <properties>
                              <property name = "hibernate.connection.url"
value="jdbc:mysql://localhost:3306/soft_uni_db?useSSL=false&amp;createDatabaseIfNotExist=true&amp;serverTimezone=UTC"/>
                              <property name = "hibernate.connection.driver_class" value="com.mysql.jdbc.Driver"/>
                              <property name = "hibernate.connection.username" value="root"/>
                              <property name = "hibernate.connection.password" value="*****"/>
                              <property name = "hibernate.dialect" value="org.hibernate.dialect.MariaDBDialect"/>
                              <property name = "hibernate.hbm2ddl.auto" value="update"/>
                              <property name = "hibernate.show_sql" value = "true" />
                              <property name = "hibernate.format_sql" value = "true" />
                    </properties>
          </persistence-unit>
</persistence>
```

Summary

- Java Servlets:
  - Servlets provide a component-based, platform-independent method for building Web based applications
- Hibernate:
  - Hibernate is an **O**bject-**R**elational **M**apping (ORM) solution for Java

Questions?


SoftUni
Software University
SoftUni Svetlina
SoftUni Creative
SoftUni Digital
SoftUni Foundation
SoftUni Kids

SoftUni Diamond Partners

Software University

XSsoftware

SBTech
we know sports

telenor

SoftwareGroup
doing it right

NETPEAK

SmartIT

Postbank
Решения за твоето утре

SUPER HOSTING .BG

INDEAVR
Serving the high achievers

INFRAGISTICS®

LIEBHERR

æternity

SoftUni Organizational Partners

Software University

ИНФОРМАЦИОННО ОБСЛУЖВАНЕ

OneBit SOFTWARE

Lukanet.com

WORLD OF MYTHS

Trainings @ Software University (SoftUni)

- Software University – High-Quality Education and Employment Opportunities
  - softuni.bg
- Software University Foundation
  - http://softuni.foundation/
- Software University @ Facebook
  - facebook.com/SoftwareUniversity
- Software University Forums
  - forum.softuni.bg

License

- This course (slides, examples, demos, videos, homework, etc.) is licensed under the "Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International" license