

Struts

Example struts can be browsed at

<https://github.com/apache/tomee/tree/master/examples/struts>

Help us document this example! Click the blue pencil icon in the upper right to edit this page.

AddUser

```
package org.superbiz.struts;

import javax.naming.Context;
import javax.naming.InitialContext;
import java.util.Properties;

public class AddUser {

    private int id;
    private String firstName;
    private String lastName;
    private String errorMessage;

    public String getFirstName() {
        return firstName;
    }

    public void setFirstName(String firstName) {
        this.firstName = firstName;
    }

    public String getLastName() {
        return lastName;
    }

    public void setLastName(String lastName) {
        this.lastName = lastName;
    }

    public String getErrorMessage() {
        return errorMessage;
    }

    public void setErrorMessage(String errorMessage) {
        this.errorMessage = errorMessage;
    }

    public int getId() {
```

```

        return id;
    }

    public void setId(int id) {
        this.id = id;
    }

    public String execute() {

        try {
            UserService service = null;
            Properties props = new Properties();
            props.put(Context.INITIAL_CONTEXT_FACTORY,
                "org.apache.openejb.core.LocalInitialContextFactory");
            Context ctx = new InitialContext(props);
            service = (UserService) ctx.lookup("UserServiceImplLocal");
            service.add(new User(id, firstName, lastName));
        } catch (Exception e) {
            this.errorMessage = e.getMessage();
            return "failure";
        }

        return "success";
    }
}

```

AddUserForm

```

package org.superbiz.struts;

import com.opensymphony.xwork2.ActionSupport;

public class AddUserForm extends ActionSupport {
}

```

FindUser

```

package org.superbiz.struts;

import javax.naming.Context;
import javax.naming.InitialContext;
import java.util.Properties;

public class FindUser {
}

```

```

private int id;
private String errorMessage;
private User user;

public User getUser() {
    return user;
}

public void setUser(User user) {
    this.user = user;
}

public String getErrorMessage() {
    return errorMessage;
}

public void setErrorMessage(String errorMessage) {
    this.errorMessage = errorMessage;
}

public int getId() {
    return id;
}

public void setId(int id) {
    this.id = id;
}

public String execute() {
    try {
        UserService service = null;
        Properties props = new Properties();
        props.put(Context.INITIAL_CONTEXT_FACTORY,
            "org.apache.openejb.core.LocalInitialContextFactory");
        Context ctx = new InitialContext(props);
        service = (UserService) ctx.lookup("UserServiceImplLocal");
        this.user = service.find(id);
    } catch (Exception e) {
        this.errorMessage = e.getMessage();
        return "failure";
    }

    return "success";
}
}

```

FindUserForm

```
package org.superbiz.struts;

import com.opensymphony.xwork2.ActionSupport;

public class FindUserForm extends ActionSupport {
}
```

ListAllUsers

```
package org.superbiz.struts;

import javax.naming.Context;
import javax.naming.InitialContext;
import java.util.List;
import java.util.Properties;

public class ListAllUsers {

    private int id;
    private String errorMessage;
    private List<User> users;

    public List<User> getUsers() {
        return users;
    }

    public void setUsers(List<User> users) {
        this.users = users;
    }

    public String getErrorMessage() {
        return errorMessage;
    }

    public void setErrorMessage(String errorMessage) {
        this.errorMessage = errorMessage;
    }

    public int getId() {
        return id;
    }

    public void setId(int id) {
        this.id = id;
    }
}
```

```

    }

    public String execute() {

        try {
            UserService service = null;
            Properties props = new Properties();
            props.put(Context.INITIAL_CONTEXT_FACTORY,
                "org.apache.openejb.core.LocalInitialContextFactory");
            Context ctx = new InitialContext(props);
            service = (UserService) ctx.lookup("UserServiceImplLocal");
            this.users = service.findAll();
        } catch (Exception e) {
            this.errorMessage = e.getMessage();
            return "failure";
        }

        return "success";
    }
}

```

User

```

package org.superbiz.struts;

import javax.persistence.Entity;
import javax.persistence.Id;
import javax.persistence.Table;
import java.io.Serializable;

@Entity
@Table(name = "USER")
public class User implements Serializable {
    private long id;
    private String firstName;
    private String lastName;

    public User(long id, String firstName, String lastName) {
        super();
        this.id = id;
        this.firstName = firstName;
        this.lastName = lastName;
    }

    public User() {
    }

    @Id
    public long getId() {
        return id;
    }

    public void setId(long id) {
        this.id = id;
    }

    public String getFirstName() {
        return firstName;
    }

    public void setFirstName(String firstName) {
        this.firstName = firstName;
    }

    public String getLastName() {
        return lastName;
    }

    public void setLastName(String lastName) {
        this.lastName = lastName;
    }
}

```

UserService

```
package org.superbiz.struts;

import java.util.List;

public interface UserService {
    public void add(User user);

    public User find(int id);

    public List<User> findAll();
}
```

UserServiceImpl

```
package org.superbiz.struts;

import javax.ejb.Stateless;
import javax.persistence.EntityManager;
import javax.persistence.PersistenceContext;
import java.util.List;

@Stateless
public class UserServiceImpl implements UserService {

    @PersistenceContext(unitName = "user")
    private EntityManager manager;

    public void add(User user) {
        manager.persist(user);
    }

    public User find(int id) {
        return manager.find(User.class, id);
    }

    public List<User> findAll() {
        return manager.createQuery("select u from User u").getResultList();
    }
}
```

persistence.xml

```
</persistence-unit>
```



```

-->
</persistence>

## struts.xml

<struts>
  <constant name="struts.devMode" value="true"></constant>
  <package name="default" namespace="/" extends="struts-default">
    <action name="addUserForm" class="org.superbiz.struts.AddUserForm">
      <result>/addUserForm.jsp</result>
    </action>
    <action name="addUser" class="org.superbiz.struts.AddUser">
      <result name="success">/addedUser.jsp</result>
      <result name='failure'>/addUserForm.jsp</result>
    </action>
    <action name="findUserForm" class="org.superbiz.struts.FindUserForm">
      <result>/findUserForm.jsp</result>
    </action>
    <action name="findUser" class="org.superbiz.struts.FindUser">
      <result name='success'>/displayUser.jsp</result>
      <result name='failure'>/findUserForm.jsp</result>
    </action>
    <action name="listAllUsers" class="org.superbiz.struts.ListAllUsers">
      <result>/displayUsers.jsp</result>
    </action>

  </package>
</struts>

## decorators.xml

<decorators defaultdir="/decorators">
  <decorator name="main" page="layout.jsp">
    <pattern>/*</pattern>
  </decorator>
</decorators>

## web.xml

<web-app xmlns="http://java.sun.com/xml/ns/javaee" xmlns:xsi=
"http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd"
  version="2.5">
  <display-name>Learn EJB3 and Struts2</display-name>
  <filter>
    <filter-name>struts2</filter-name>
    <filter-class>org.apache.struts2.dispatcher.FilterDispatcher</filter-class>
    <init-param>
      <param-name>actionPackages</param-name>

```

```

    <param-value>com.lq</param-value>
  </init-param>
</filter>
<filter>
  <filter-name>struts-cleanup</filter-name>
  <filter-class>org.apache.struts2.dispatcher.ActionContextCleanUp</filter-class>
</filter>
<filter>
  <filter-name>sitemesh</filter-name>
  <filter-class>com.opensymphony.module.sitemesh.filter.PageFilter</filter-class>
</filter>
<filter-mapping>
  <filter-name>struts-cleanup</filter-name>
  <url-pattern>/*</url-pattern>
</filter-mapping>
<filter-mapping>
  <filter-name>sitemesh</filter-name>
  <url-pattern>/*</url-pattern>
</filter-mapping>
<filter-mapping>
  <filter-name>struts2</filter-name>
  <url-pattern>/*</url-pattern>
</filter-mapping>
<welcome-file-list>
  <welcome-file>index.jsp</welcome-file>
</welcome-file-list>
<jsp-config>
  <jsp-property-group>
    <description>JSP configuration of all the JSP's</description>
    <url-pattern>*.jsp</url-pattern>
    <include-prelude>/prelude.jspf</include-prelude>
  </jsp-property-group>
</jsp-config>
</web-app>

```