

# Component Interfaces

Example component-interfaces can be browsed at <https://github.com/apache/tomee/tree/master/examples/component-interfaces>

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

## FriendlyPerson

```
package org.superbiz;

import javax.ejb.Init;
import javax.ejb.Local;
import javax.ejb.LocalHome;
import javax.ejb.Remote;
import javax.ejb.RemoteHome;
import javax.ejb.Remove;
import javax.ejb.Stateful;
import java.text.MessageFormat;
import java.util.HashMap;
import java.util.Locale;
import java.util.Properties;

/**
 * This is an EJB 3 style pojo stateful session bean
 * it does not need to implement javax.ejb.SessionBean
 *
 */
//START SNIPPET: code

// EJB 3.0 Style business interfaces
// Each of these interfaces are already annotated in the classes
// themselves with @Remote and @Local, so annotating them here
// in the bean class again is not really required.
@Remote({FriendlyPersonRemote.class})
@Local({FriendlyPersonLocal.class})

// EJB 2.1 Style component interfaces
// These interfaces, however, must be annotated here in the bean class.
// Use of @RemoteHome in the FriendlyPersonEjbHome class itself is not allowed.
// Use of @LocalHome in the FriendlyPersonEjbLocalHome class itself is also not
// allowed.
@RemoteHome(FriendlyPersonEjbHome.class)
@LocalHome(FriendlyPersonEjbLocalHome.class)

@Stateful
public class FriendlyPerson implements FriendlyPersonLocal, FriendlyPersonRemote {

    private final HashMap<String, MessageFormat> greetings;
```

```

private final Properties languagePreferences;

private String defaultLanguage;

public FriendlyPerson() {
    greetings = new HashMap();
    languagePreferences = new Properties();
    defaultLanguage = Locale.getDefault().getLanguage();

    addGreeting("en", "Hello {0}!");
    addGreeting("es", "Hola {0}!");
    addGreeting("fr", "Bonjour {0}!");
    addGreeting("pl", "Witaj {0}!");
}

/**
 * This method corresponds to the FriendlyPersonEjbHome.create() method
 * and the FriendlyPersonEjbLocalHome.create()
 * <p/>
 * If you do not have an EJBHome or EJBLocalHome interface, this method
 * can be deleted.
 */
@Init
public void create() {
}

/**
 * This method corresponds to the following methods:
 * - EJBObject.remove()
 * - EJBHome.remove(ejbObject)
 * - EJBLocalObject.remove()
 * - EJBLocalHome.remove(ejbObject)
 * <p/>
 * If you do not have an EJBHome or EJBLocalHome interface, this method
 * can be deleted.
 */
@Remove
public void remove() {
}

public String greet(String friend) {
    String language = languagePreferences.getProperty(friend, defaultLanguage);
    return greet(language, friend);
}

public String greet(String language, String friend) {
    MessageFormat greeting = greetings.get(language);
    if (greeting == null) {
        Locale locale = new Locale(language);
        return "Sorry, I don't speak " + locale.getDisplayLanguage() + ".";
    }
}

```

```

        return greeting.format(new Object[]{friend});
    }

    public void addGreeting(String language, String message) {
        greetings.put(language, new MessageFormat(message));
    }

    public void setLanguagePreferences(String friend, String language) {
        languagePreferences.put(friend, language);
    }

    public String getDefaultLanguage() {
        return defaultLanguage;
    }

    public void setDefaultLanguage(String defaultLanguage) {
        this.defaultLanguage = defaultLanguage;
    }
}

```

## FriendlyPersonEjbHome

```

package org.superbiz;

//START SNIPPET: code

import javax.ejb.CreateException;
import javax.ejb.EJBHome;
import java.rmi.RemoteException;

public interface FriendlyPersonEjbHome extends EJBHome {
    FriendlyPersonEjbObject create() throws CreateException, RemoteException;
}

```

## FriendlyPersonEjbLocalHome

```
package org.superbiz;

//START SNIPPET: code

import javax.ejb.CreateException;
import javax.ejb.EJBLocalHome;
import java.rmi.RemoteException;

public interface FriendlyPersonEjbLocalHome extends EJBLocalHome {
    FriendlyPersonEjbLocalObject create() throws CreateException, RemoteException;
}
```

## FriendlyPersonEjbLocalObject

```
package org.superbiz;

import javax.ejb.EJBLocalObject;

public interface FriendlyPersonEjbLocalObject extends EJBLocalObject {
    String greet(String friend);

    String greet(String language, String friend);

    void addGreeting(String language, String message);

    void setLanguagePreferences(String friend, String language);

    String getDefaultLanguage();

    void setDefaultLanguage(String defaultLanguage);
}
```

## FriendlyPersonEjbObject

```

package org.superbiz;

//START SNIPPET: code

import javax.ejb.EJBObject;
import java.rmi.RemoteException;

public interface FriendlyPersonEjbObject extends EJBObject {
    String greet(String friend) throws RemoteException;

    String greet(String language, String friend) throws RemoteException;

    void addGreeting(String language, String message) throws RemoteException;

    void setLanguagePreferences(String friend, String language) throws
RemoteException;

    String getDefaultLanguage() throws RemoteException;

    void setDefaultLanguage(String defaultLanguage) throws RemoteException;
}

```

## FriendlyPersonLocal

```

package org.superbiz;

//START SNIPPET: code

import javax.ejb.Local;

@Local
public interface FriendlyPersonLocal {
    String greet(String friend);

    String greet(String language, String friend);

    void addGreeting(String language, String message);

    void setLanguagePreferences(String friend, String language);

    String getDefaultLanguage();

    void setDefaultLanguage(String defaultLanguage);
}

```

# FriendlyPersonRemote

```
package org.superbiz;

import javax.ejb.Remote;

//START SNIPPET: code
@Remote
public interface FriendlyPersonRemote {
    String greet(String friend);

    String greet(String language, String friend);

    void addGreeting(String language, String message);

    void setLanguagePreferences(String friend, String language);

    String getDefaultLanguage();

    void setDefaultLanguage(String defaultLanguage);
}
```

# FriendlyPersonTest

```
package org.superbiz;

import junit.framework.TestCase;

import javax.ejb.embeddable.EJBContainer;
import javax.naming.Context;
import java.util.Locale;

/**
 * @version $Rev: 1090810 $ $Date: 2011-04-10 07:49:26 -0700 (Sun, 10 Apr 2011) $
 */
public class FriendlyPersonTest extends TestCase {

    private Context context;

    protected void setUp() throws Exception {
        context = EJBContainer.createEJBContainer().getContext();
    }

    /**
     * Here we lookup and test the FriendlyPerson bean via its EJB 2.1 EJBHome and
     * EJBObject interfaces
     */
}
```

```

* @throws Exception
*/
//START SNIPPET: remotehome
public void testEjbHomeAndEjbObject() throws Exception {
    Object object = context.lookup("java:global/component-
interfaces/FriendlyPerson!org.superbiz.FriendlyPersonEjbHome");
    FriendlyPersonEjbHome home = (FriendlyPersonEjbHome) object;
    FriendlyPersonEjbObject friendlyPerson = home.create();

    friendlyPerson.setDefaultLanguage("en");

    assertEquals("Hello David!", friendlyPerson.greet("David"));
    assertEquals("Hello Amelia!", friendlyPerson.greet("Amelia"));

    friendlyPerson.setLanguagePreferences("Amelia", "es");

    assertEquals("Hello David!", friendlyPerson.greet("David"));
    assertEquals("Hola Amelia!", friendlyPerson.greet("Amelia"));

    // Amelia took some French, let's see if she remembers
    assertEquals("Bonjour Amelia!", friendlyPerson.greet("fr", "Amelia"));

    // Dave should take some Polish and if he had, he could say Hi in Polish
    assertEquals("Witaj Dave!", friendlyPerson.greet("pl", "Dave"));

    // Let's see if I speak Portuguese
    assertEquals("Sorry, I don't speak " + new Locale("pt").getDisplayLanguage() +
    ".", friendlyPerson.greet("pt", "David"));

    // Ok, well I've been meaning to learn, so...
    friendlyPerson.addGreeting("pt", "Ola {0}!");

    assertEquals("Ola David!", friendlyPerson.greet("pt", "David"));
}
//END SNIPPET: remotehome

/**
 * Here we lookup and test the FriendlyPerson bean via its EJB 2.1 EJBLocalHome
and EJBLocalObject interfaces
 *
 * @throws Exception
 */
public void testEjbLocalHomeAndEjbLocalObject() throws Exception {
    Object object = context.lookup("java:global/component-
interfaces/FriendlyPerson!org.superbiz.FriendlyPersonEjbLocalHome");
    FriendlyPersonEjbLocalHome home = (FriendlyPersonEjbLocalHome) object;
    FriendlyPersonEjbLocalObject friendlyPerson = home.create();

    friendlyPerson.setDefaultLanguage("en");

```



```

assertEquals("Hello David!", friendlyPerson.greet("David"));
assertEquals("Hello Amelia!", friendlyPerson.greet("Amelia"));

friendlyPerson.setLanguagePreferences("Amelia", "es");

assertEquals("Hello David!", friendlyPerson.greet("David"));
assertEquals("Hola Amelia!", friendlyPerson.greet("Amelia"));

// Amelia took some French, let's see if she remembers
assertEquals("Bonjour Amelia!", friendlyPerson.greet("fr", "Amelia"));

// Dave should take some Polish and if he had, he could say Hi in Polish
assertEquals("Witaj Dave!", friendlyPerson.greet("pl", "Dave"));

// Let's see if I speak Portuguese
assertEquals("Sorry, I don't speak " + new Locale("pt").getDisplayLanguage() +
".", friendlyPerson.greet("pt", "David"));

// Ok, well I've been meaning to learn, so...
friendlyPerson.addGreeting("pt", "Ola {0}!");

assertEquals("Ola David!", friendlyPerson.greet("pt", "David"));
}

/**
 * Here we lookup and test the FriendlyPerson bean via its EJB 3.0 business remote
interface
 *
 * @throws Exception
 */
//START SNIPPET: remote
public void testBusinessRemote() throws Exception {
    Object object = context.lookup("java:global/component-
interfaces/FriendlyPerson!org.superbiz.FriendlyPersonRemote");

    FriendlyPersonRemote friendlyPerson = (FriendlyPersonRemote) object;

    friendlyPerson.setDefaultLanguage("en");

    assertEquals("Hello David!", friendlyPerson.greet("David"));
    assertEquals("Hello Amelia!", friendlyPerson.greet("Amelia"));

    friendlyPerson.setLanguagePreferences("Amelia", "es");

    assertEquals("Hello David!", friendlyPerson.greet("David"));
    assertEquals("Hola Amelia!", friendlyPerson.greet("Amelia"));

    // Amelia took some French, let's see if she remembers
    assertEquals("Bonjour Amelia!", friendlyPerson.greet("fr", "Amelia"));

    // Dave should take some Polish and if he had, he could say Hi in Polish

```

```

    assertEquals("Witaj Dave!", friendlyPerson.greet("pl", "Dave"));

    // Let's see if I speak Portuguese
    assertEquals("Sorry, I don't speak " + new Locale("pt").getDisplayLanguage() +
        ".", friendlyPerson.greet("pt", "David"));

    // Ok, well I've been meaning to learn, so...
    friendlyPerson.addGreeting("pt", "Ola {0}!");

    assertEquals("Ola David!", friendlyPerson.greet("pt", "David"));
}
//START SNIPPET: remote

/**
 * Here we lookup and test the FriendlyPerson bean via its EJB 3.0 business local
 * interface
 *
 * @throws Exception
 */
public void testBusinessLocal() throws Exception {
    Object object = context.lookup("java:global/component-
        interfaces/FriendlyPerson!org.superbiz.FriendlyPersonLocal");

    FriendlyPersonLocal friendlyPerson = (FriendlyPersonLocal) object;

    friendlyPerson.setDefaultLanguage("en");

    assertEquals("Hello David!", friendlyPerson.greet("David"));
    assertEquals("Hello Amelia!", friendlyPerson.greet("Amelia"));

    friendlyPerson.setLanguagePreferences("Amelia", "es");

    assertEquals("Hello David!", friendlyPerson.greet("David"));
    assertEquals("Hola Amelia!", friendlyPerson.greet("Amelia"));

    // Amelia took some French, let's see if she remembers
    assertEquals("Bonjour Amelia!", friendlyPerson.greet("fr", "Amelia"));

    // Dave should take some Polish and if he had, he could say Hi in Polish
    assertEquals("Witaj Dave!", friendlyPerson.greet("pl", "Dave"));

    // Let's see if I speak Portuguese
    assertEquals("Sorry, I don't speak " + new Locale("pt").getDisplayLanguage() +
        ".", friendlyPerson.greet("pt", "David"));

    // Ok, well I've been meaning to learn, so...
    friendlyPerson.addGreeting("pt", "Ola {0}!");

    assertEquals("Ola David!", friendlyPerson.greet("pt", "David"));
}

```

# Running

## ----- T E S T S -----

```
Running org.superbiz.FriendlyPersonTest
Apache OpenEJB 4.0.0-beta-1    build: 20111002-04:06
http://tomee.apache.org/
INFO - openejb.home = /Users/dblevins/examples/component-interfaces
INFO - openejb.base = /Users/dblevins/examples/component-interfaces
INFO - Using 'javax.ejb.embeddable.EJBContainer=true'
INFO - Configuring Service(id=Default Security Service, type=SecurityService,
provider-id=Default Security Service)
INFO - Configuring Service(id=Default Transaction Manager, type=TransactionManager,
provider-id=Default Transaction Manager)
INFO - Found EjbModule in classpath: /Users/dblevins/examples/component-
interfaces/target/classes
INFO - Beginning load: /Users/dblevins/examples/component-interfaces/target/classes
INFO - Configuring enterprise application: /Users/dblevins/examples/component-
interfaces
INFO - Configuring Service(id=Default Stateful Container, type=Container, provider-
id=Default Stateful Container)
INFO - Auto-creating a container for bean FriendlyPerson: Container(type=STATEFUL,
id=Default Stateful Container)
INFO - Configuring Service(id=Default Managed Container, type=Container, provider-
id=Default Managed Container)
INFO - Auto-creating a container for bean org.superbiz.FriendlyPersonTest:
Container(type=MANAGED, id=Default Managed Container)
INFO - Enterprise application "/Users/dblevins/examples/component-interfaces" loaded.
INFO - Assembling app: /Users/dblevins/examples/component-interfaces
INFO - Jndi(name="java:global/component-
interfaces/FriendlyPerson!org.superbiz.FriendlyPersonLocal")
INFO - Jndi(name="java:global/component-
interfaces/FriendlyPerson!org.superbiz.FriendlyPersonRemote")
INFO - Jndi(name="java:global/component-
interfaces/FriendlyPerson!org.superbiz.FriendlyPersonEjbLocalHome")
INFO - Jndi(name="java:global/component-
interfaces/FriendlyPerson!org.superbiz.FriendlyPersonEjbHome")
INFO - Jndi(name="java:global/component-interfaces/FriendlyPerson")
INFO -
Jndi(name="java:global/EjbModule803660549/org.superbiz.FriendlyPersonTest!org.superbiz
.FriendlyPersonTest")
INFO - Jndi(name="java:global/EjbModule803660549/org.superbiz.FriendlyPersonTest")
INFO - Created Ejb(deployment-id=FriendlyPerson, ejb-name=FriendlyPerson,
container=Default Stateful Container)
INFO - Created Ejb(deployment-id=org.superbiz.FriendlyPersonTest, ejb-
```

```
name=org.superbiz.FriendlyPersonTest, container=Default Managed Container)
INFO - Started Ejb(deployment-id=FriendlyPerson, ejb-name=FriendlyPerson,
container=Default Stateful Container)
INFO - Started Ejb(deployment-id=org.superbiz.FriendlyPersonTest, ejb-
name=org.superbiz.FriendlyPersonTest, container=Default Managed Container)
INFO - Deployed Application(path=/Users/dblevins/examples/component-interfaces)
INFO - EJBContainer already initialized. Call ejbContainer.close() to allow
reinitialization
INFO - EJBContainer already initialized. Call ejbContainer.close() to allow
reinitialization
INFO - EJBContainer already initialized. Call ejbContainer.close() to allow
reinitialization
Tests run: 4, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 1.444 sec
```

Results :

Tests run: 4, Failures: 0, Errors: 0, Skipped: 0