



SOFTWARE ENGINEERING 2 GLASSFISH

Inspection document version 1.0

Politecnico di Milano A.A. 2015-2016

Marija Mavcheva (mat. 838647)

AndrijanaMirchevska (mat. 838622)

05/01/2016

Table of Contents

1. Assigned class.....	3
2. Functional role of assigned class.....	3
3. Inspection of the code	4
3.1 Method 1: createNamingContext().....	4
3.2 Method 2: addEnvironment.....	6

1. Assigned class

The methods that were assigned to our group are: **createNamingContext()** and **addEnvironment(ContextEnvironmentenv)**. They are under the NamingContextListener class which belongs to the package org.apache.catalina.core.

2. Functional role of assigned class

NamingContextListener

The functional roles of the methods that were assigned to our group are the following:

createNamingContext()

It throws naming exceptions for the context and resources

- Creates subcontext
- Sets a new naming resource from the NamingResources() class if the initialized naming resource is null
- Add ResourceLink
- Add Resource
- Add environment
- Add EJB
- Bind user transaction
- Bind resources

addEnvironment(ContextEnvironmentenv)

It checks of what type of the environment is and then adds it as an environment entry. It tries the different types of values for env.getValue() and throws an error message if it's not of any valid type and it catches the exception when the environment entry has invalid value.

3. Inspection of the code

3.1 Method 1: createNamingContext()

Naming Conventions:

1.
 - Line 689: parameter compCtx does not give enough information about what it represents
 - Line 690: parameter envCtx does not give enough information about what it represents
 - Line 748: rb does not give enough information about what it represents
 - Line 759: rb does not give enough information about what it represents
2.
 - Line 688: container represents a class and should be with upper case

Braces

10. "Kernighan and Ritchie" style is used for the bracing

11.
 - Line 698: The if statement should be in curly braces. Open brace in line 698, after the if condition, and closing brace in line 700. So the code should be:

```
if (log.isLoggable(Level.FINE)) {  
    log.log(Level.FINE, "Creating JNDI naming context");  
}
```

File Organization

13.
 - Line 707 and 708: Should be in one line since the line length does not exceed 80 characters
 - Line 728 and 729: Should be in one line since the line length does not exceed 80 characters
 - Line 745 and 746: Should be in one line since the line length does not exceed 80 characters
 - Line 756 and 757: should be in one line since the line length does not exceed 80 characters

Wrapping Lines

15.
 - Line 745: There`s a break line after a function

Comments

18.

- Line 706: insufficient comment
- Line 713: insufficient comment
- Line 719: insufficient comment
- Line 727: insufficient comment

Java Source Files

23.

- Line 685: NamingException not specified in Javadoc
- Line 748: MessageFormat not specified in Javadoc

Initialization and Declarations

29.

- Line 696: inti is not declared in the proper scope. It can be declared inside all the for loops where it is used

31.

- Line 702: namingResources should be declared and initialized before

3.2 Method 2: addEnvironment

Naming Conventions:

1.

- Line 849: rb does not give enough information about what it represents
- Line 853: rb does not give enough information about what it represents
- Line 856: rb does not give enough information about what it represents
- Line 865: parameter envCtx does not give enough information about what it represents
- Line 866: parameter envCtx does not give enough information about what it represents

Indentation

8.

- Line 864: there is a space after the opening ``
- Line 865: indentation not ok, 4 spaces are needed
- Line 866: indentation not ok, 4 spaces are needed

Braces

10.

- Line 863: there should be an open brace after if condition and closing brace in line 867

Comments

18.

- Line 794 and 795: comment should be in one line and before line 793

Java Source Files

23.

- Line 867: NamingException not specified in Javadoc
- Line 849, 853, 856, 868: MessageFormat not specified in Javadoc

Exceptions

52.

- Line 852 and Line 855: the two catches can be written as one using || (OR) since they give the same message. So the code will be the following:

```
catch (NumberFormatException e) || (IllegalArgumentException e){  
    String msg = MessageFormat.format(rb.getString(ENV_ENTRY_INVALID_VALUE),  
env.getName());  
    log(msg);  
}
```