Code Coverage
Outlines

- Code Coverage.
- EMMA
- Installing EMMA
- Running EMMA
- View Result
Code Coverage

- Is a measure used in software testing. It describes the degree to which the source code of a program has been tested.
- Coverage analysis attempts to address questions about when to stop testing, or the amount of testing that is enough for a given program.
Code Coverage

- Code Coverage analysis allows determining the completeness of the test cases, and the percentage of the code exercised by executing the test cases.
Coverage Criteria

- To measure how well the program is exercised by a **test suite**, one or more **coverage criteria** are used.
  - Statement Coverage
  - Branch Coverage
  - Condition Coverage
  - Path Coverage
  - Function Coverage
EMMA

- EMMA is a free code coverage tool.
- Supported coverage types: `class`, `method`, `line`, `basic block`.
- EMMA can detect when a single source code line is covered only partially.
Netbeans Plugin for EMMA

- The functionality provided by the plugin helps to visually (and quickly) identify the portions of Java code with low coverage and helps in targeted tests development.
Feature

- Java sources coloring according to the coverage information from the latest unit tests execution.
- Automated java code markup updated after running unit tests or reopening file
Feature

- Code coverage markup info displayed at the java editor sidebar
- Currently Java Application, Java Library, Java Project with Existing Sources and NetBeans module projects are supported.
EMMA Plugin in Use

- You need to develop your JUnit Test.
- Make sure that Coverage Plugin is installed.
- Activate the coverage collection action.
- view coverage reports.
Install Code Coverage Plugin
Install Code Coverage Plugin

1. Open the NetBeans IDE.
2. Go to the Tools menu and select 'Plugins'.
3. In the 'Available Plugins' tab, search for 'Code Coverage Plugin'.
4. Install the plugin by checking the checkbox next to 'Code Coverage Plugin'.
5. Restart NetBeans IDE to make sure the plugin is loaded.

The Code Coverage Plugin is a NetBeans 6.0 plugin that enhances the existing NetBeans functionality with new code coverage features. The plugin works as an additional service that tracks all files according to the unit tests and coverage information. With code coverage plugin enabled, users can work with their project in the usual way but can easily view the test coverage of the project classes. The code coverage plugin will update the code coverage data and refresh editors markup every time a unit test is executed for the project. Currently, the Java Application, Java Library, Java Project with Existing Sources and NetBeans modules projects are supported.
Activate Coverage Collection
Execute Your JUnit Test Code
View Coverage Reports
Coverage Report
Java sources coloring according to the coverage information
Improve Your JWP Coverage

- Avoid (remove) unreachable code (function, block, statement).
- Implement JUnit that cover GUI components.
- Modify JWP classes if needed to improve your test.
# Code Coverage with Unreachable Code

![Screenshot of Code Coverage Tool](image)

### Code Coverage Details

- **Project Coverage**
  - Total classes covered: 100% (1/1)
  - Total lines covered: 90% (60/60)
  - Total packages covered: 100% (1/1)

### Package Coverage

<table>
<thead>
<tr>
<th>Package Name</th>
<th>Classes</th>
<th>Lines</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test</td>
<td>100% (1/1)</td>
<td>90% (60/60)</td>
</tr>
</tbody>
</table>

### Class Coverage

<table>
<thead>
<tr>
<th>Class Name</th>
<th>Lines</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test.MainframeTest</td>
<td>100% (60/60)</td>
</tr>
</tbody>
</table>

### Test Results

- All tests passed.
- Test. MainframeTest passed.

```java
// Code snippet
```