University of Sheffield

ICSME 2024

40th International Conference on Software Maintenance and Evolution

Flagstaff, AZ, USA
Oct 6 - Oct 11, 2024

PseudoSweep
A Pseudo-Tested Code Identifier

Megan Maton, Gregory M. Kapfhammer, Phil McMinn

PseudoSweep

Finds Pseudo-tested Code in **Java 8–11** Projects

Use with Gradle or Maven Projects

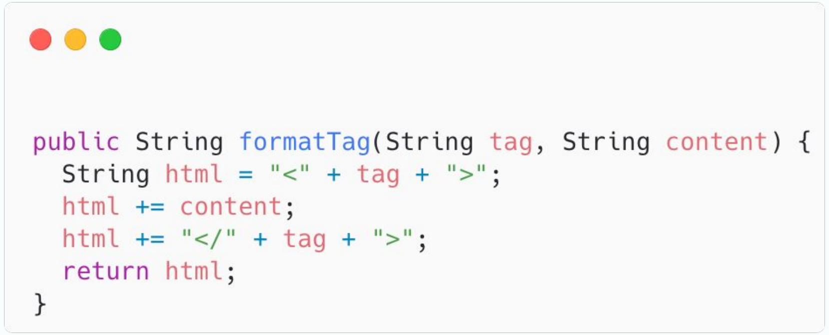
https://maven.apache.org/
https://docs.gradle.org/current/userguide/userguide.html

Pseudo-tested Code

Code that is executed by the test suite, yet when removed, the test suite continues to pass.

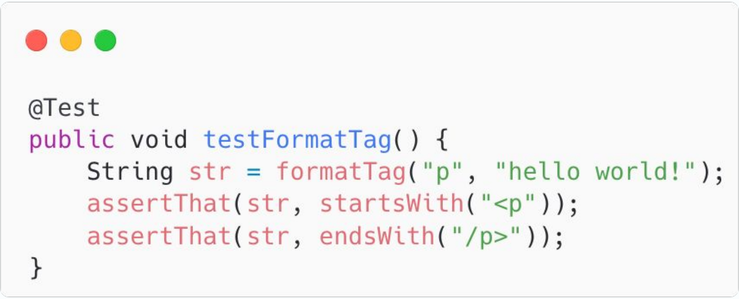
Identifying pseudo-tested code

Production Code



```java
public String formatTag(String tag, String content) {
  String html = "<" + tag + ">";
  html += content;
  html += "</" + tag + ">";
  return html;
}
```

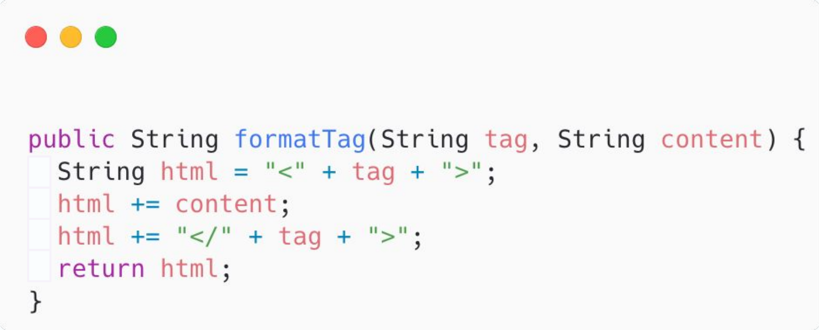
Test Suite



```java
@Test
public void testFormatTag() {
    String str = formatTag("p", "hello world!");
    assertThat(str, startsWith("<p"));
    assertThat(str, endsWith("/p>"));
}
```

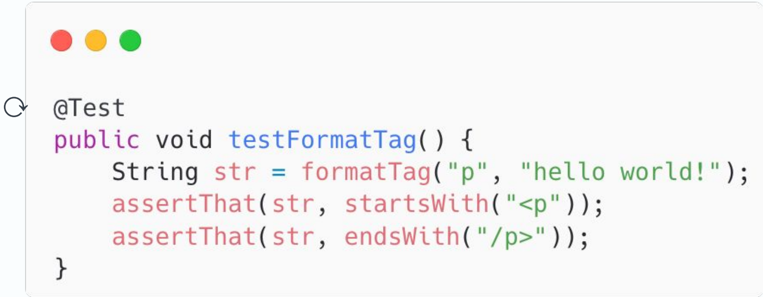
Identifying pseudo-tested code

Production Code



```
public String formatTag(String tag, String content) {
  String html = "<" + tag + ">";
  html += content;
  html += "</" + tag + ">";
  return html;
}
```

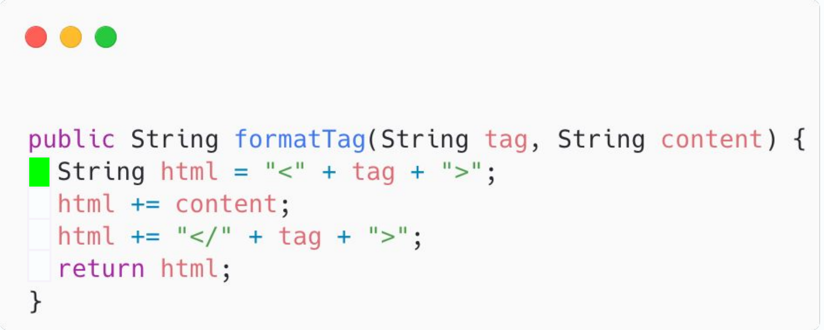
Test Suite



```
@Test
public void testFormatTag() {
    String str = formatTag("p", "hello world!");
    assertThat(str, startsWith("<p"));
    assertThat(str, endsWith("/p>"));
}
```

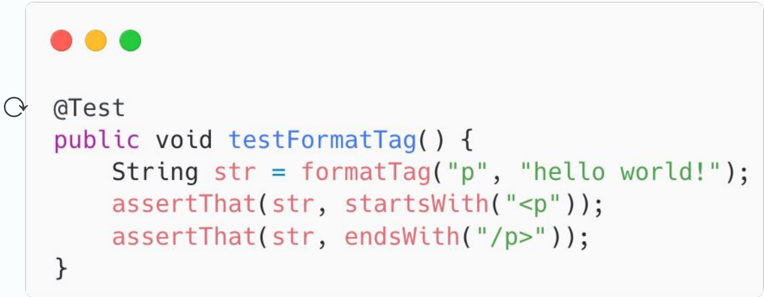
Identifying pseudo-tested code

Production Code



```java
public String formatTag(String tag, String content) {
  String html = "<" + tag + ">";
  html += content;
  html += "</" + tag + ">";
  return html;
}
```

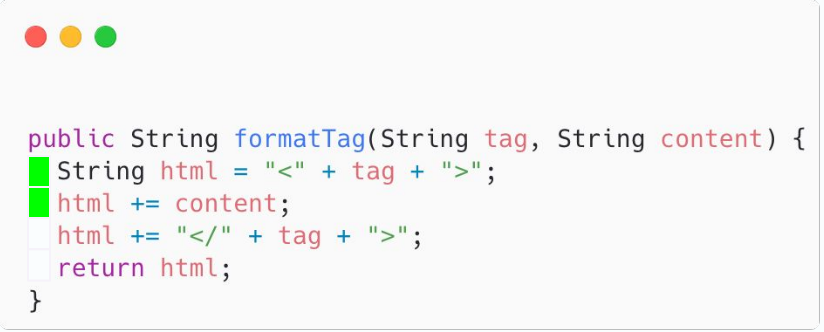
Test Suite



```java
@Test
public void testFormatTag() {
    String str = formatTag("p", "hello world!");
    assertThat(str, startsWith("<p"));
    assertThat(str, endsWith("/p>"));
}
```

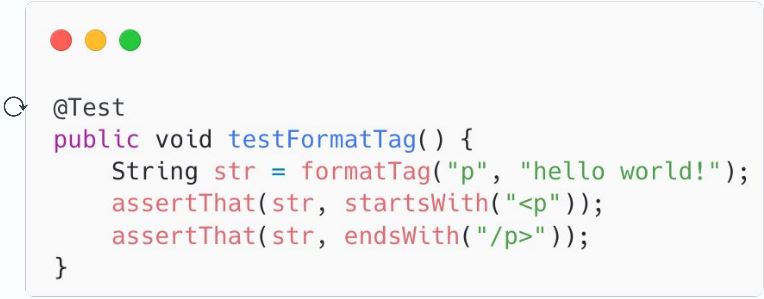
Identifying pseudo-tested code

Production Code



```java
public String formatTag(String tag, String content) {
 String html = "<" + tag + ">";
 html += content;
 html += "</" + tag + ">";
 return html;
}
```

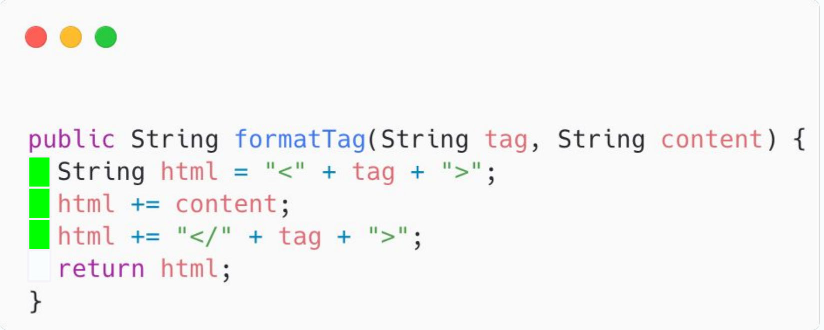
Test Suite



```java
@Test
public void testFormatTag() {
    String str = formatTag("p", "hello world!");
    assertThat(str, startsWith("<p"));
    assertThat(str, endsWith("/p>"));
}
```

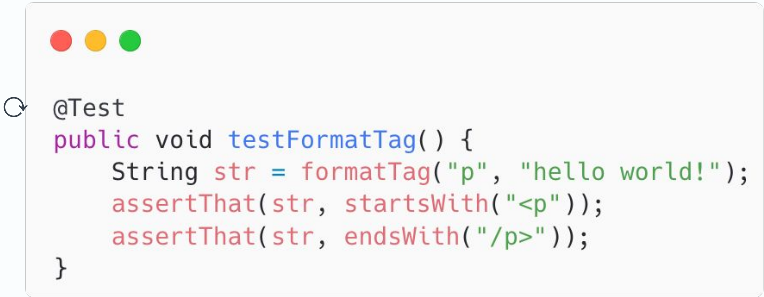
Identifying pseudo-tested code

Production Code



```java
public String formatTag(String tag, String content) {
  String html = "<" + tag + ">";
  html += content;
  html += "</" + tag + ">";
  return html;
}
```

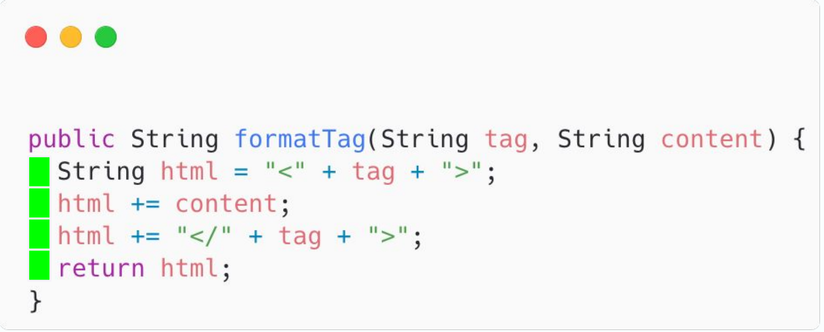
Test Suite



```java
@Test
public void testFormatTag() {
    String str = formatTag("p", "hello world!");
    assertThat(str, startsWith("<p"));
    assertThat(str, endsWith("/p>"));
}
```

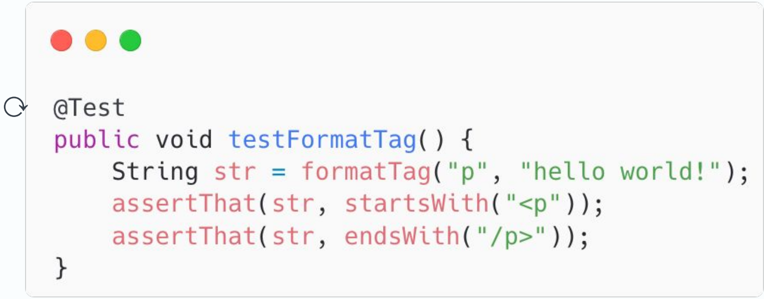
Identifying pseudo-tested code

Production Code



```
public String formatTag(String tag, String content) {
🟩String html = "<" + tag + ">";
🟩html += content;
🟩html += "</" + tag + ">";
🟩return html;
}
```

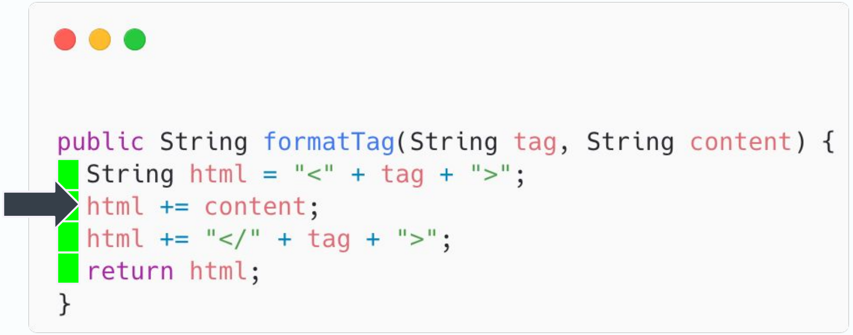
Test Suite



```
@Test
public void testFormatTag() {
    String str = formatTag("p", "hello world!");
    assertThat(str, startsWith("<p"));
    assertThat(str, endsWith("/p>"));
}
```

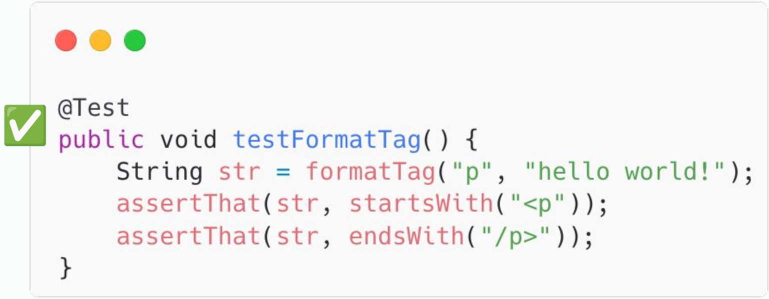
Identifying pseudo-tested code

Production Code



```
public String formatTag(String tag, String content) {
  String html = "<" + tag + ">";
  html += content;
  html += "</" + tag + ">";
  return html;
}
```

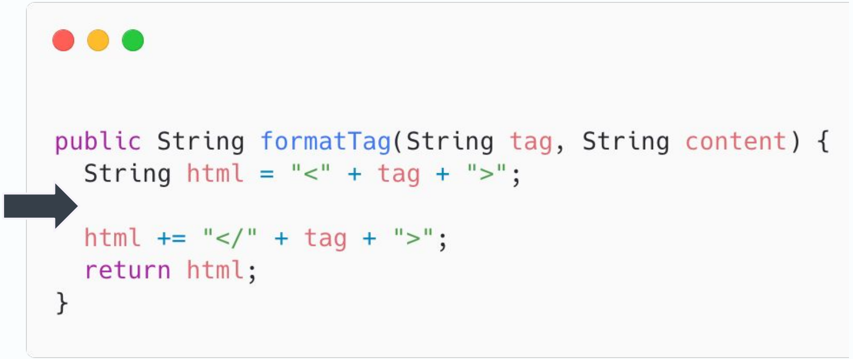
Test Suite



```
✅ @Test
public void testFormatTag() {
    String str = formatTag("p", "hello world!");
    assertThat(str, startsWith("<p"));
    assertThat(str, endsWith("/p>"));
}
```

Identifying pseudo-tested code

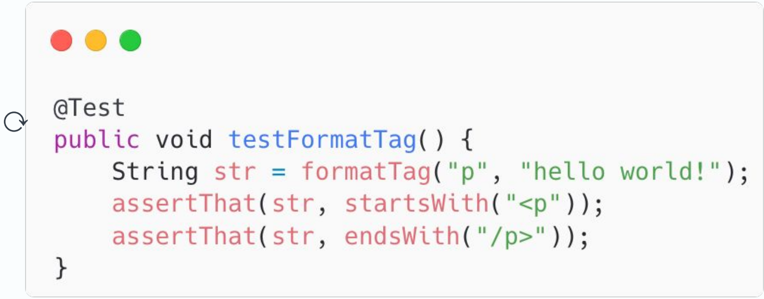
Production Code



```
public String formatTag(String tag, String content) {
  String html = "<" + tag + ">";

  html += "</" + tag + ">";
  return html;
}
```

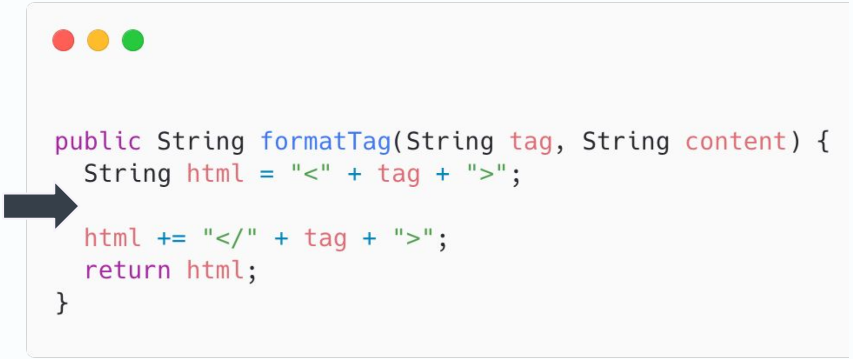
Test Suite



```
@Test
public void testFormatTag() {
    String str = formatTag("p", "hello world!");
    assertThat(str, startsWith("<p"));
    assertThat(str, endsWith("/p>"));
}
```

Identifying pseudo-tested code

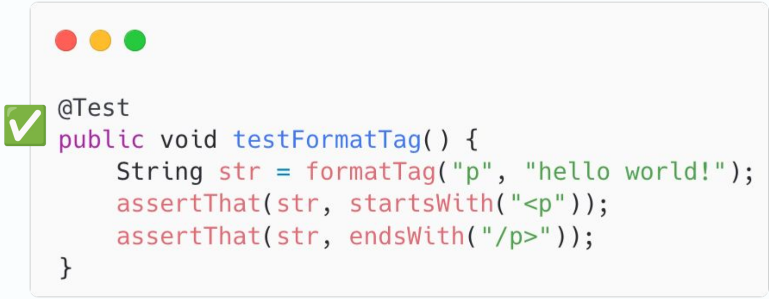
Production Code



```
public String formatTag(String tag, String content) {
  String html = "<" + tag + ">";

  html += "</" + tag + ">";
  return html;
}
```

Test Suite

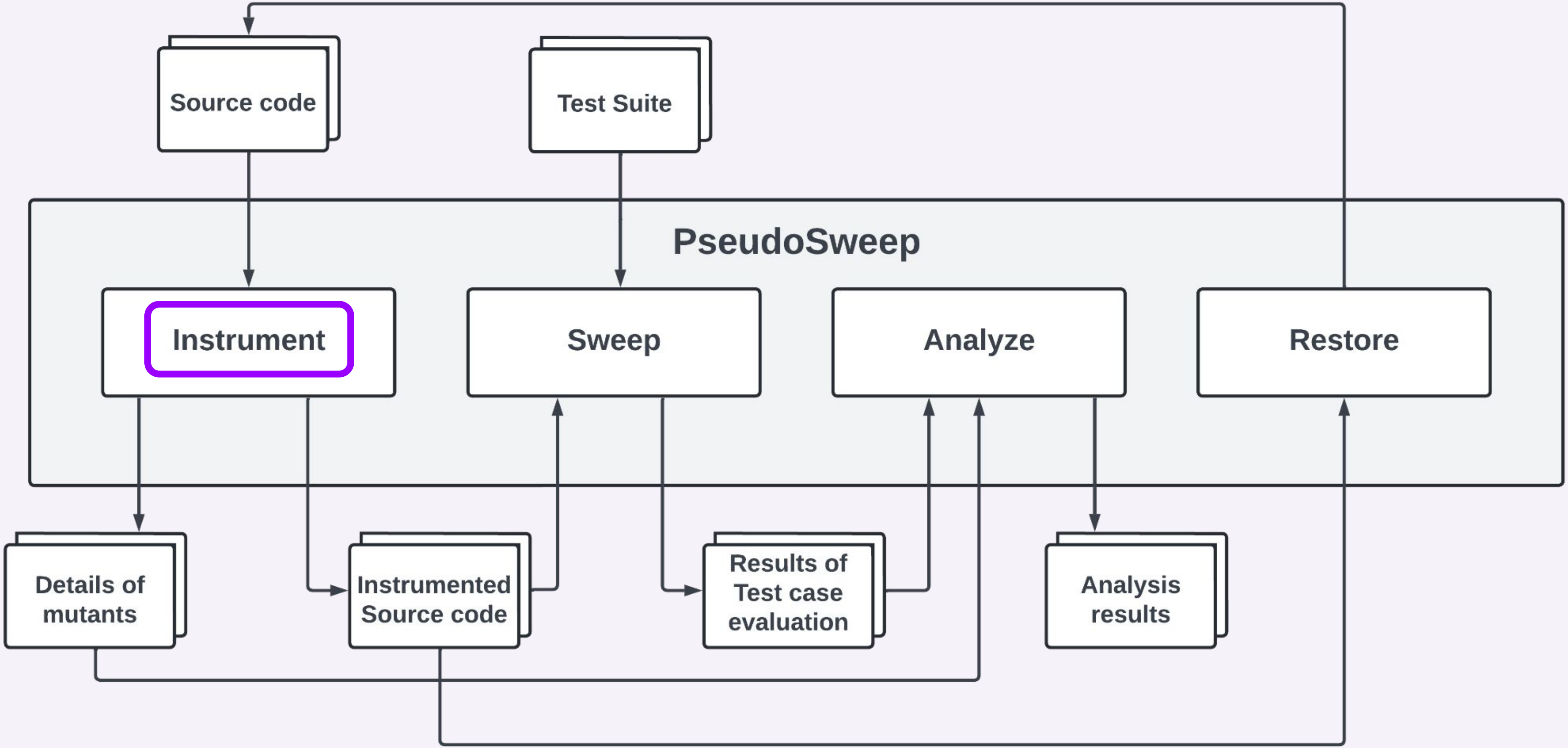


```
✅ @Test
public void testFormatTag() {
       String str = formatTag("p", "hello world!");
       assertThat(str, startsWith("<p"));
       assertThat(str, endsWith("/p>"));
}
```

The test suite didn’t notice the missing statement…

PseudoSweep reveals these pseudo-tested statements

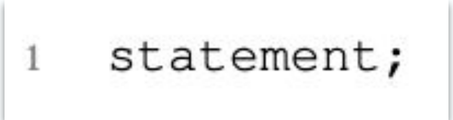
Using PseudoSweep


Source code
Test Suite
PseudoSweep
Instrument
Sweep
Analyze
Restore
Details of mutants
Instrumented Source code
Results of Test case evaluation
Analysis results

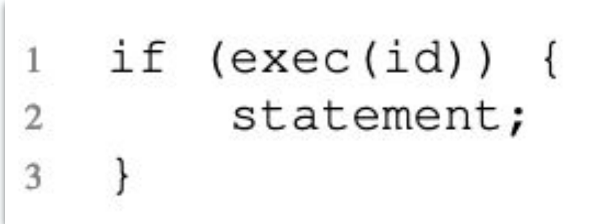
1. Instrument

Original

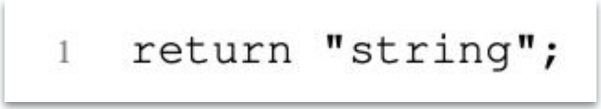
Instrumented



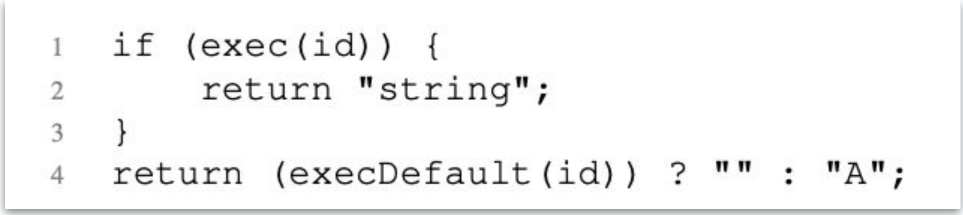
```
statement;
```



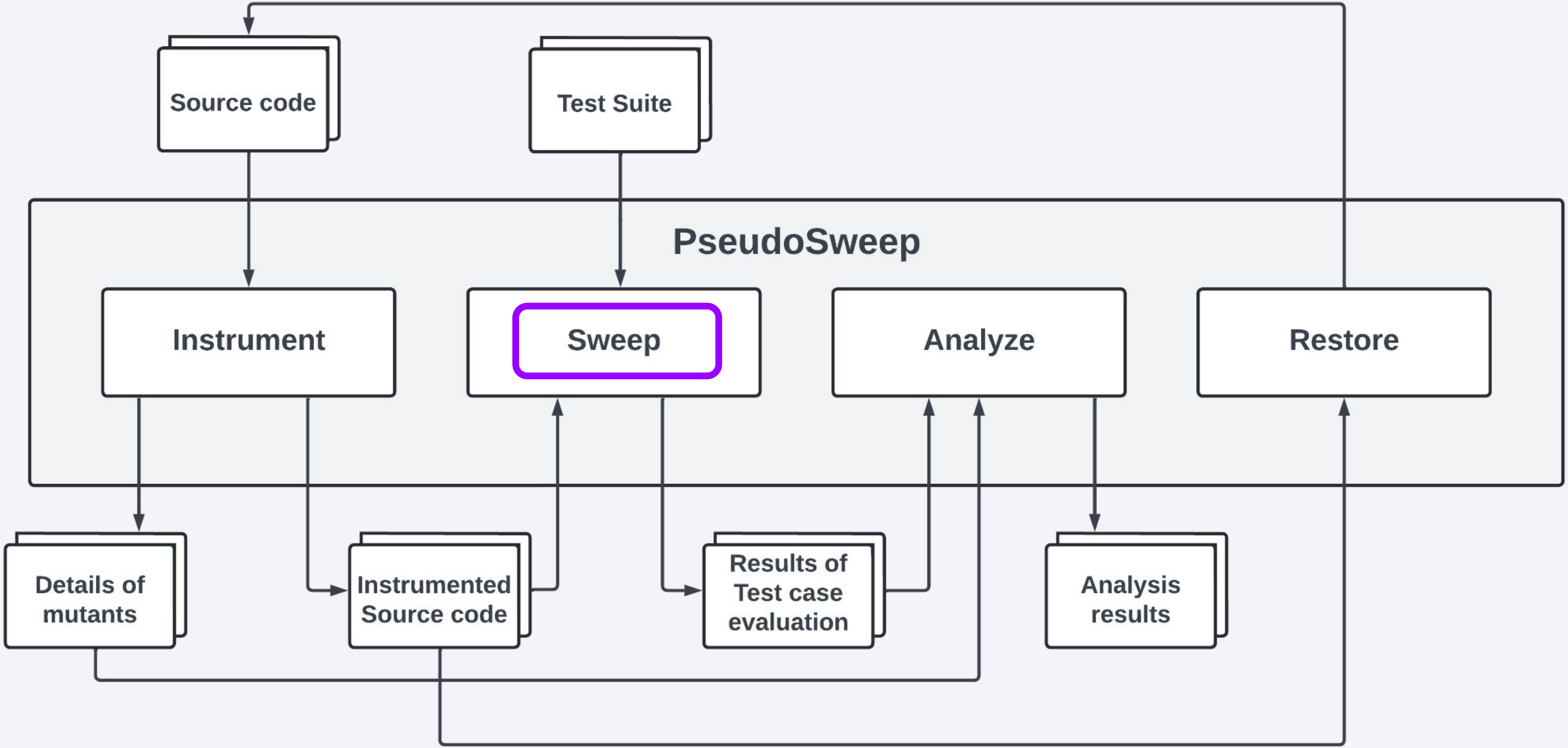
```
if (exec(id)) {
      statement;
}
```



```
return "string";
```

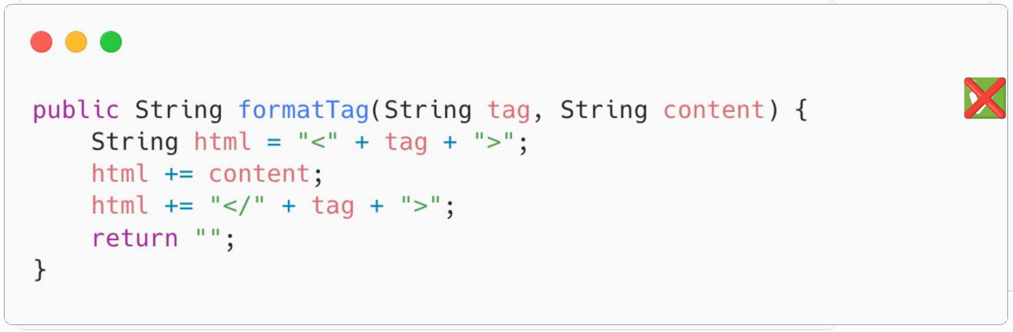


```
if (exec(id)) {
      return "string";
}
return (execDefault(id)) ? "" : "A";
```


Source code
Test Suite
PseudoSweep
Instrument
Sweep
Analyze
Restore
Details of mutants
Instrumented Source code
Results of Test case evaluation
Analysis results

2. Sweep

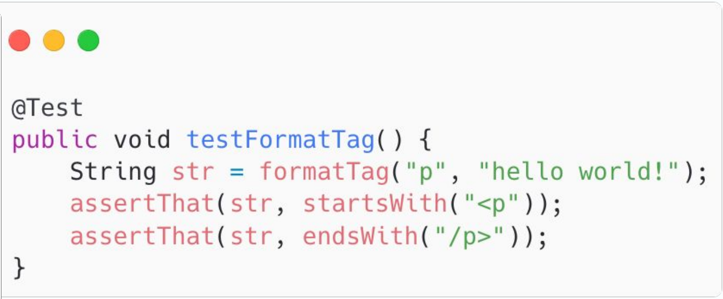
Production Code



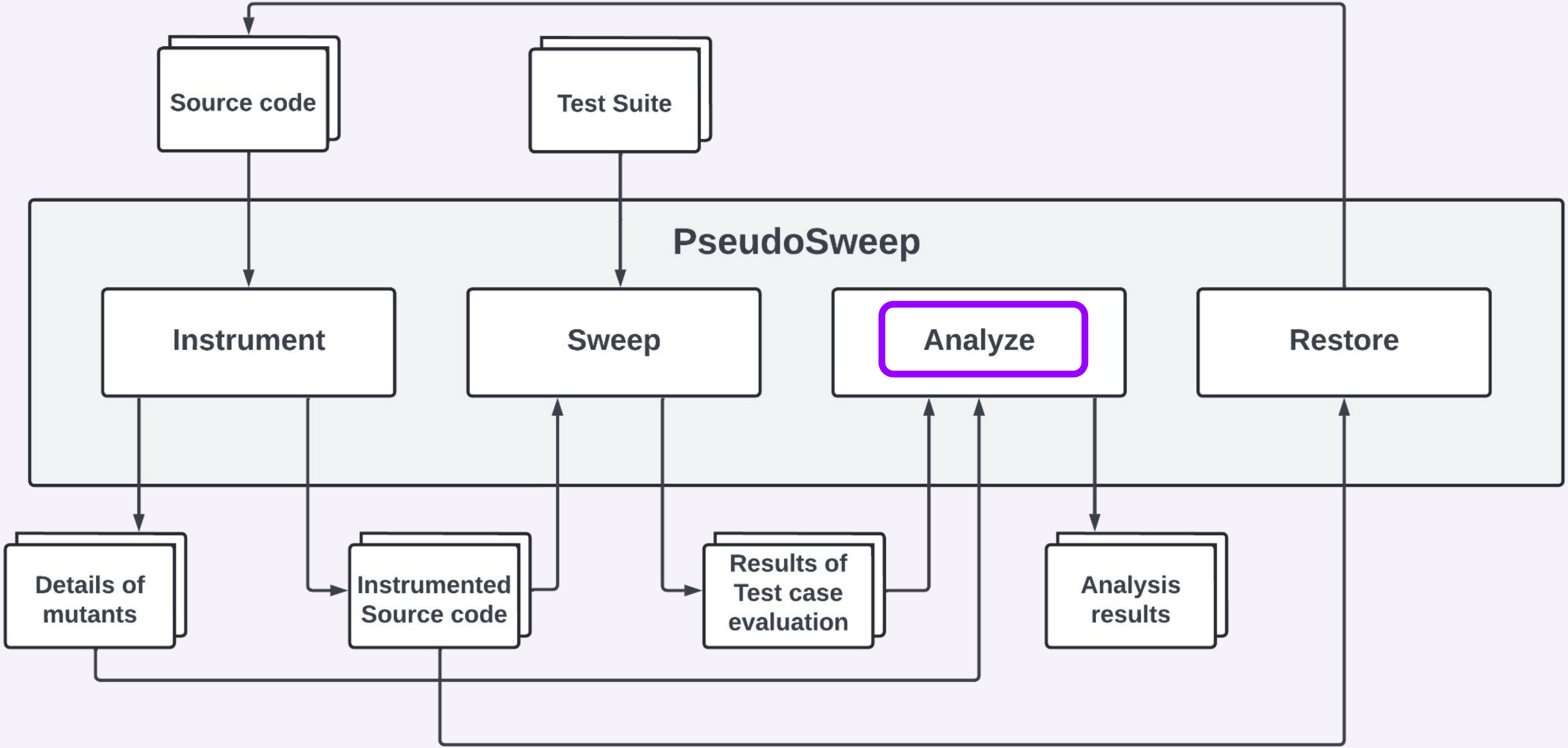
```java
public String formatTag(String tag, String content) {
    String html = "<" + tag + ">";
    html += content;
    html += "</" + tag + ">";
    return "";
}
```

❌

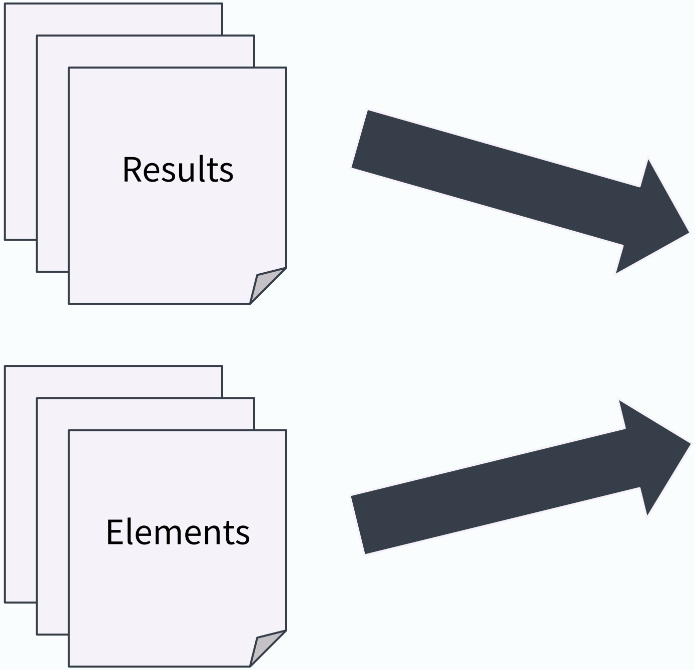
Test Suite

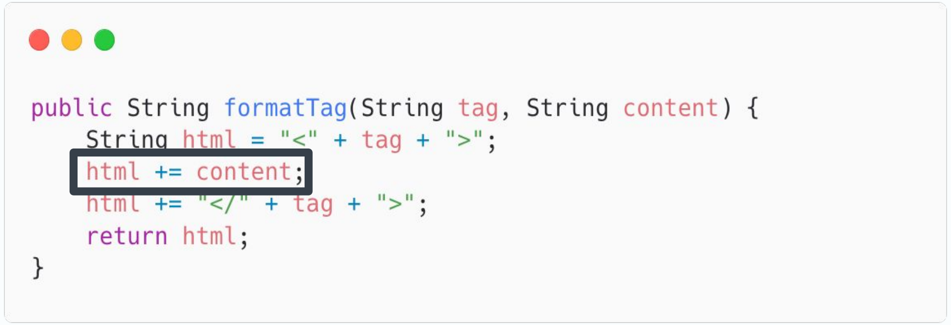


```java
@Test
public void testFormatTag() {
    String str = formatTag("p", "hello world!");
    assertThat(str, startsWith("<p"));
    assertThat(str, endsWith("/p>"));
}
```

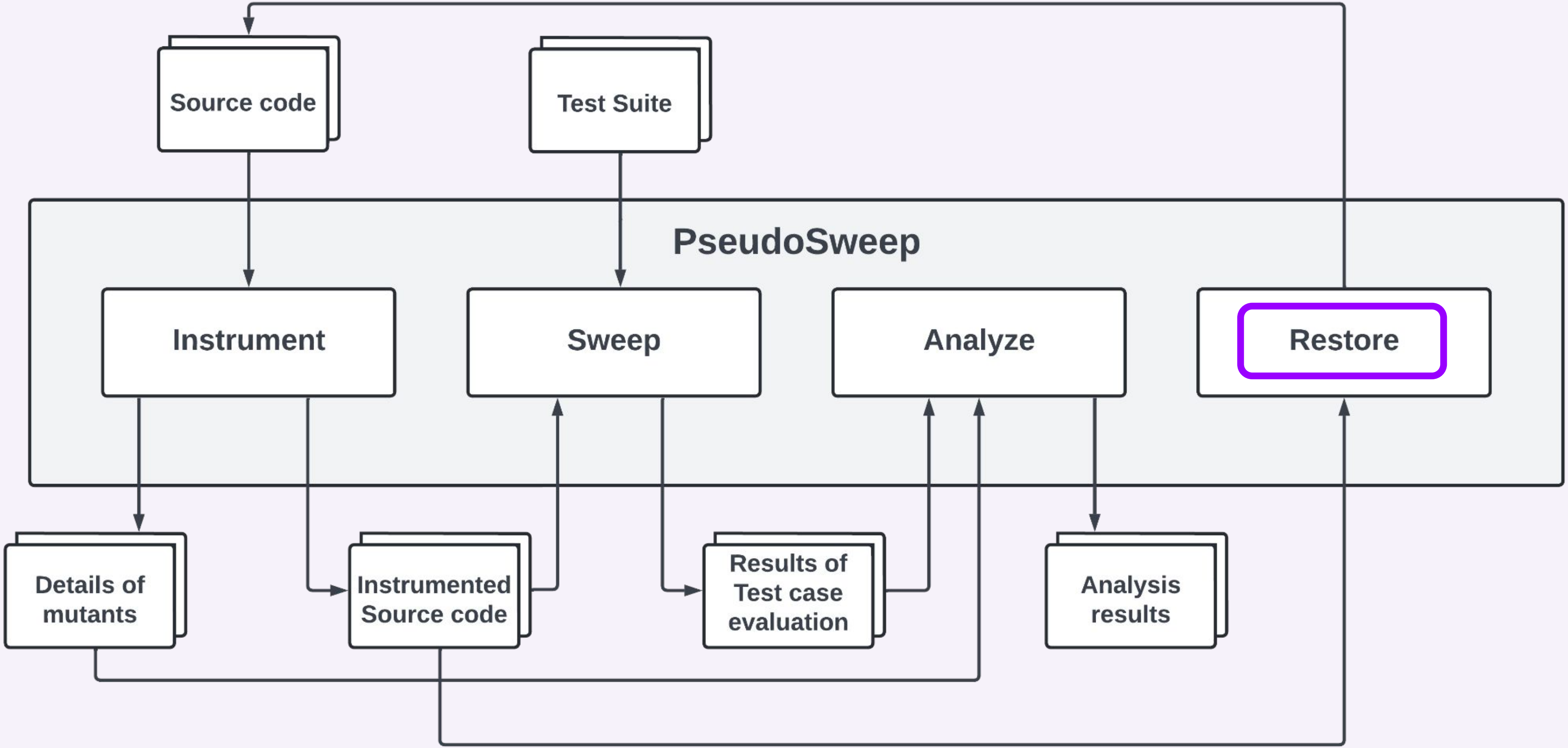

Source code
Test Suite
PseudoSweep
Instrument
Sweep
Analyze
Restore
Details of mutants
Instrumented Source code
Results of Test case evaluation
Analysis results

3. Analyze



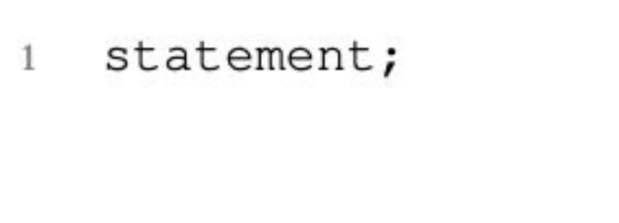


```
public String formatTag(String tag, String content) {
    String html = "<" + tag + ">";
    html += content;
    html += "</" + tag + ">";
    return html;
}
```


Source code
Test Suite
PseudoSweep
Instrument
Sweep
Analyze
Restore
Details of mutants
Instrumented Source code
Results of Test case evaluation
Analysis results

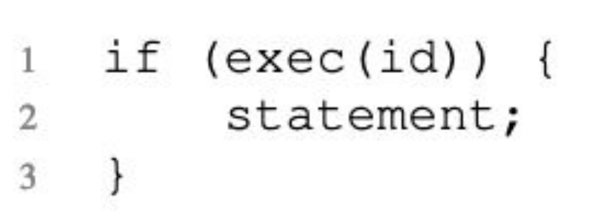
4. Restore

Original



```
statement;
```

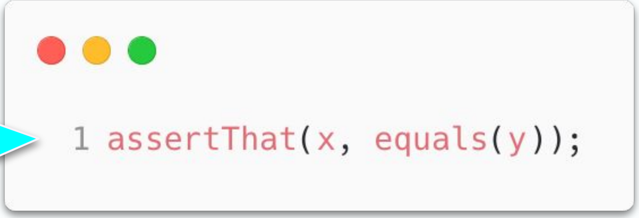
Instrumented



```
if (exec(id)) {
        statement;
}
```

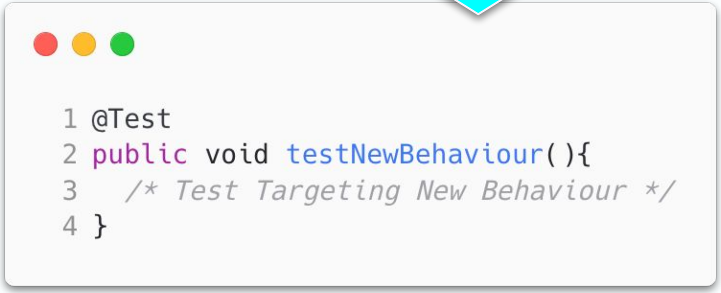
Solutions

Add assertions



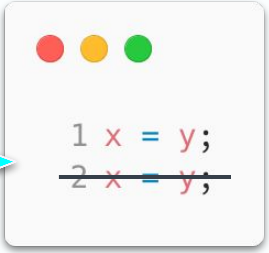
```
assertThat(x, equals(y));
```

Add behaviour specific tests



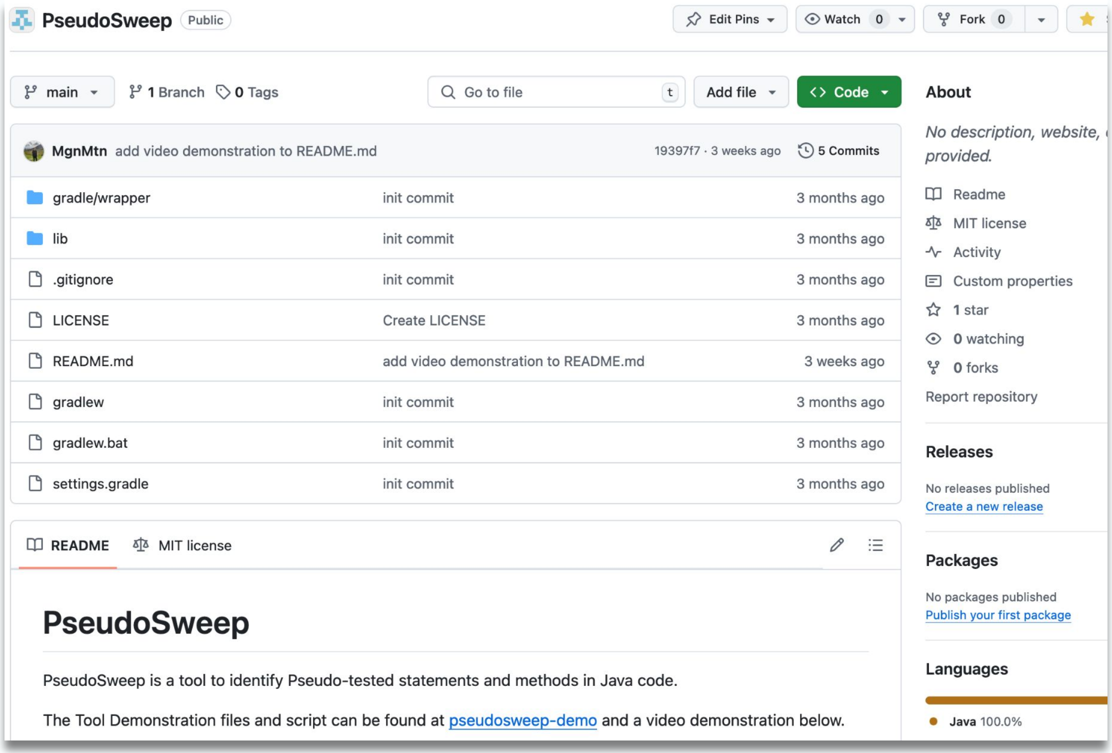
```
@Test
public void testNewBehaviour(){
  /* Test Targeting New Behaviour */
}
```

Remove redundant code



```
x = y;
~~x = y;~~
```

https://github.com/PseudoTested/PseudoSweep



MgnMtn add video demonstration to README.md 19397f7 · 3 weeks ago 5 Commits

gradle/wrapper	init commit	3 months ago
lib	init commit	3 months ago
.gitignore	init commit	3 months ago
LICENSE	Create LICENSE	3 months ago
README.md	add video demonstration to README.md	3 weeks ago
gradlew	init commit	3 months ago
gradlew.bat	init commit	3 months ago
settings.gradle	init commit	3 months ago

No description, website, provided.

No releases published
Create a new release

No packages published
Publish your first package

PseudoSweep

PseudoSweep is a tool to identify Pseudo-tested statements and methods in Java code.

The Tool Demonstration files and script can be found at pseudosweep-demo and a video demonstration below.