Assertions

Debugging Strategies

Sophie Engle ◆ sjengle@cs.usfca.edu
Department of Computer Science

Introduction

- Makes programming assumptions (hopes?) explicit
  - Assuming input in private method is never null

- Can embed throughout code
  - No output produced if assertion is correct

- Can enable/disable assertions via compiler flag
  - Add –ea flag as vm argument when running

http://docs.oracle.com/javase/8/docs/technotes/guides/language/assert.html

CS 212 Software Development
Spring 2017

Sophie Engle ◆ sjengle@cs.usfca.edu
Department of Computer Science

UNIVERSITY OF
SAN FRANCISCO

Using Assertions

Remember, assertions can be disabled!

- Do not use for input validation in public methods

- Do not use assertions to perform actions necessary for functionality of program

http://docs.oracle.com/javase/8/docs/technotes/guides/language/assert.html

CS 212 Software Development
Spring 2017

Sophie Engle ◆ sjengle@cs.usfca.edu
Department of Computer Science

UNIVERSITY OF
SAN FRANCISCO

Use for Internal Invariants

```
1. if (i % 3 == 0) {
2.     ...
3. } else if (i % 3 == 1) {
4.     ...
5. } else {
6.     assert i % 3 == 2;
7.     ...
8. }
```

```
1. switch (value) {
2. case 0:
3.     ...
4. case 1:
5.     ...
6. default:
7.     assert false;
8. }
```

http://docs.oracle.com/javase/8/docs/technotes/guides/language/assert.html

Use for Control Invariants

```
1  void firstFound(List<String> elements, String find) {
2    for (String element : elements) {
3      if (element.equals(find)) {
4        System.out.println("Found: " + element);
5        return;
6      }
7    }
8    // know element is in list, should never reach this point
9    assert false;
10 }
```

http://docs.oracle.com/javase/8/docs/technotes/guides/language/assert.html

CS 212 Software Development
Spring 2017

Sophie Engle ◆ sjengle@cs.usfca.edu
Department of Computer Science

UNIVERSITY OF
SAN FRANCISCO

Use for Private Preconditions

```
1    public void addWord(String word) {
2        if (word != null) {
3            words.add(convertString(word));
4        }
5    }
6
7    private String convertString(String text) {
8        assert text != null;
9        ...
10       return text;
11   }
```

http://docs.oracle.com/javase/8/docs/technotes/guides/language/assert.html

CS 212 Software Development
Spring 2017

Sophie Engle ◆ sjengle@cs.usfca.edu
Department of Computer Science

UNIVERSITY OF
SAN FRANCISCO

Use for Postconditions

```
1  public long copy(Path source, Path destination) {
2    try {
3        copyFile(source, destination);
4    } catch (Exception ex) {
5        return -1;
6    }
7    assert Files.size(source) == Files.size(destination);
8    return Files.size(destination);
9  }
```

http://docs.oracle.com/javase/8/docs/technotes/guides/language/assert.html

CS 212 Software Development
Spring 2017

Sophie Engle ◆ sjengle@cs.usfca.edu
Department of Computer Science

UNIVERSITY OF
SAN FRANCISCO

Summary

- Do use to indicate any assumptions made by programmer

- Do use to verify input in private methods

- Do NOT use to validate user input in public methods

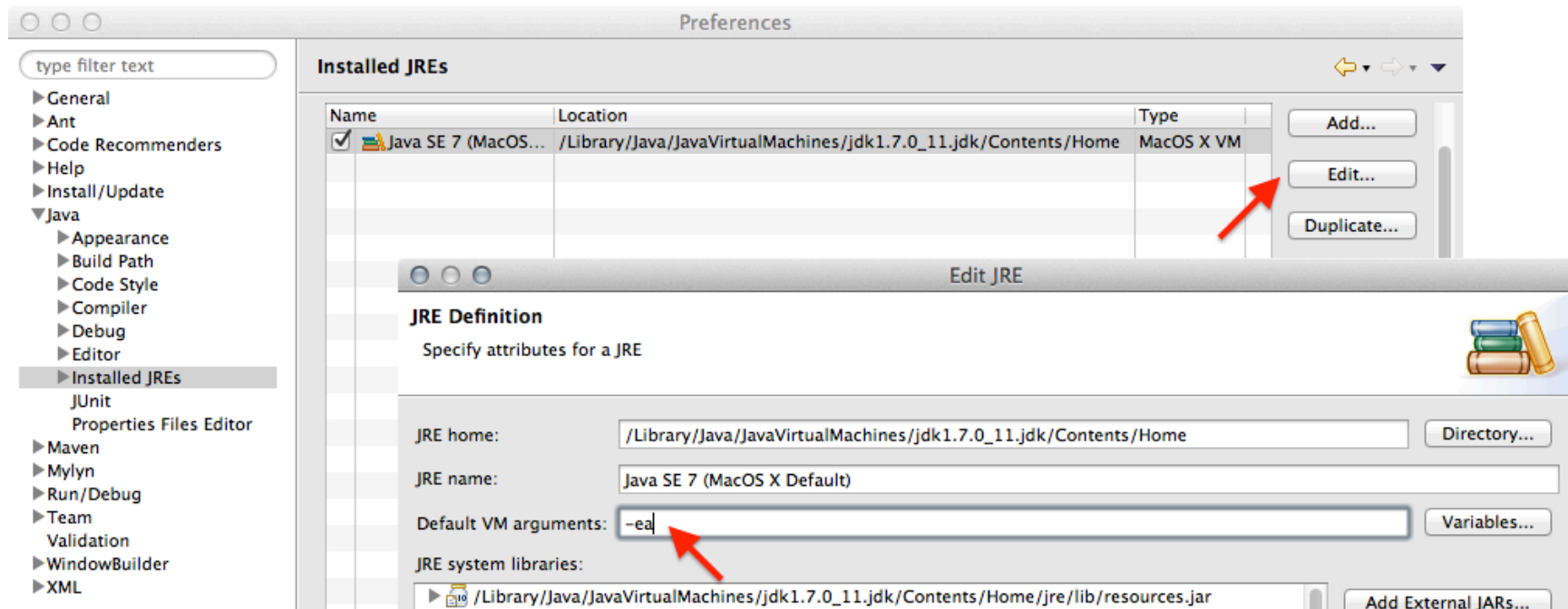- Do NOT use to perform necessary operations

http://docs.oracle.com/javase/8/docs/technotes/guides/language/assert.html

CS 212 Software Development
Spring 2017

Sophie Engle ◆ sjengle@cs.usfca.edu
Department of Computer Science

UNIVERSITY OF
SAN FRANCISCO

Enabling Assertions

- Assertions are not enabled by default
  - Require –ea (enable assertions) flag when running

- Two methods for enabling in Eclipse
  - Always enable assertions for all Java projects
  - Only enable assertions for certain Java Projects
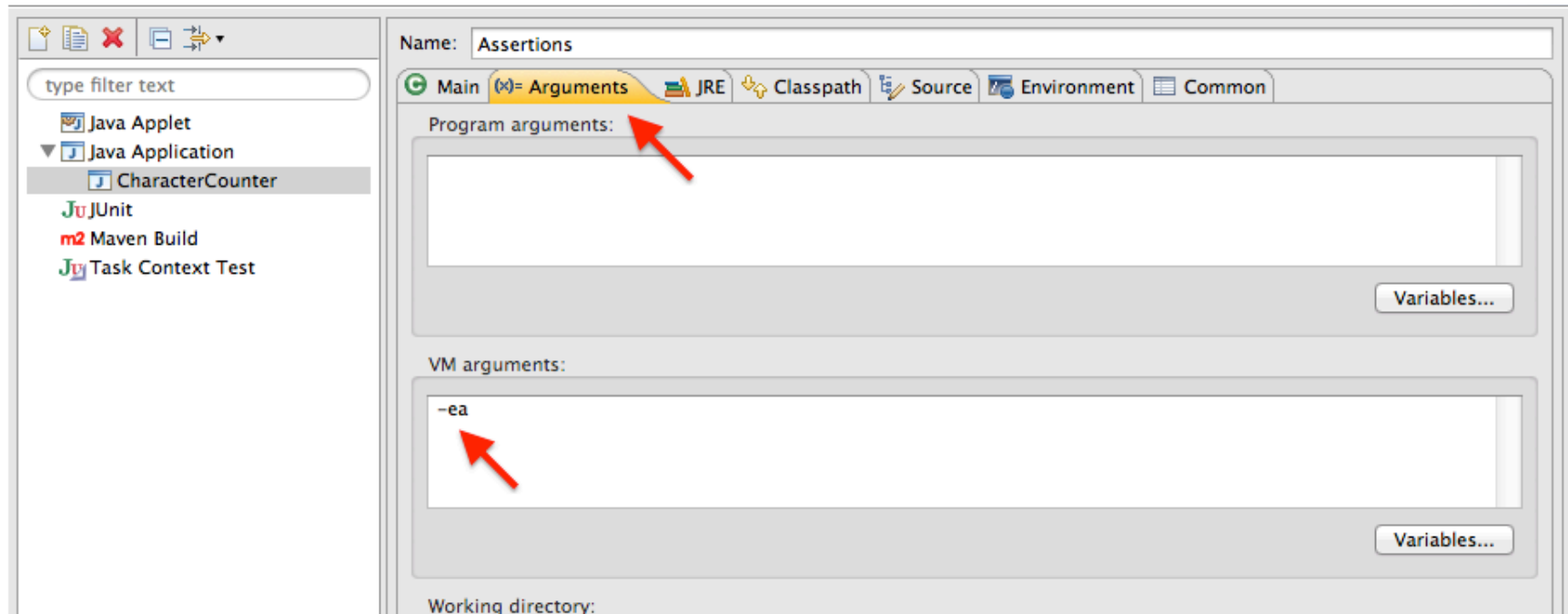
http://tutoringcenter.cs.usfca.edu/resources/enabling-assertions-in-eclipse.html

CS 212 Software Development
Spring 2017

Sophie Engle ◆ sjengle@cs.usfca.edu
Department of Computer Science

UNIVERSITY OF
SAN FRANCISCO

Always Enable Assertions

Preferences » Installed JREs » Java SE 8 » Edit… » Default VM arguments » -ea » Finish

CS 212 Software Development
Spring 2017

Sophie Engle ◆ sjengle@cs.usfca.edu
Department of Computer Science

UNIVERSITY OF
SAN FRANCISCO

Enable Pre-Project Assertions

Run Configurations » Arguments » VM arguments » -ea » Run

CS 212 Software Development
Spring 2017

Sophie Engle • sjengle@cs.usfca.edu
Department of Computer Science

UNIVERSITY OF
SAN FRANCISCO

Sophie Engle ◆ sjengle@cs.usfca.edu
Department of Computer Science