

A Beginner's Guide to Project 1

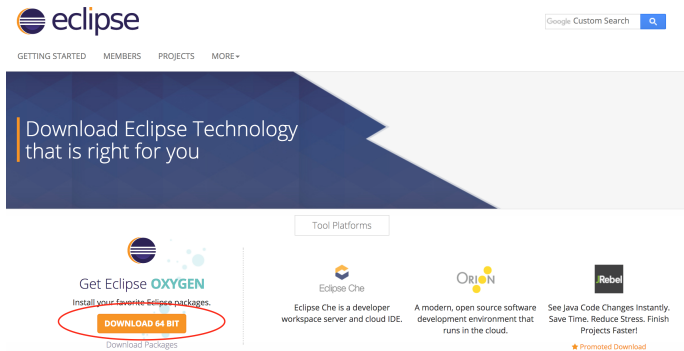
11-711: Algorithms for NLP

Fall 2019

- ➊ Install the Required Software
- ➋ Getting Started with Your Project
- ➌ Eclipse Basic Usages
- ➍ Navigating the Starter Code
- ➎ Your Work
- ➏ Java Basics

- 1 Install the Required Software
- 2 Getting Started with Your Project
- 3 Eclipse Basic Usages
- 4 Navigating the Starter Code
- 5 Your Work
- 6 Java Basics

- Link: <https://www.eclipse.org/downloads/>



- Follow their instructions

- May create unexpected results with `HashMap`, `HashSet`, etc.
- What you see on your computers may be different from what we see when grading
- Do this in your Terminal

```
$ getconf LONG_BIT
```

- Most of your Mac computers are compatible with 64 bits.

- On Mac

```
$ /usr/bin/ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"  
$ brew install ant
```

- On Linux

```
$ sudo apt-get update  
$ sudo apt-get install ant
```

- On Windows

- DIY: <https://www.mkyong.com/ant/how-to-install-apache-ant-on-windows/>

- 1 Install the Required Software
- 2 Getting Started with Your Project**
- 3 Eclipse Basic Usages
- 4 Navigating the Starter Code
- 5 Your Work
- 6 Java Basics

- Starter code:

`https://storage.googleapis.com/11711/code1.tar.gz`

- Data:

`https://storage.googleapis.com/11711/data1.tar.gz`

- Unzip

```
$ tar -xzf code1.tar.gz
```

```
$ tar -xzf data1.tar.gz
```


- Go to code1

```
$ ant -f build_assign1.xml
```

```
$ export DATA_PATH="/absolute/path/to/your/data/folder"
```

```
$ java -cp assign1.jar:assign1-submit.jar -server -mx500m \  
  edu.berkeley.nlp.assignments.assign1.LanguageModelTester \  
  -path $DATA_PATH -lmType UNIGRAM
```

- Create a `run.sh` script that launches your experiment

```
$ vim run.sh
```

```
$ # copy the scripts from the previous slide
```

```
$ # changing UNIGRAM to TRIGRAM
```

```
$ chmod 777 run.sh
```

- Now you can run your experiment with 1 command

```
$ ./run.sh
```

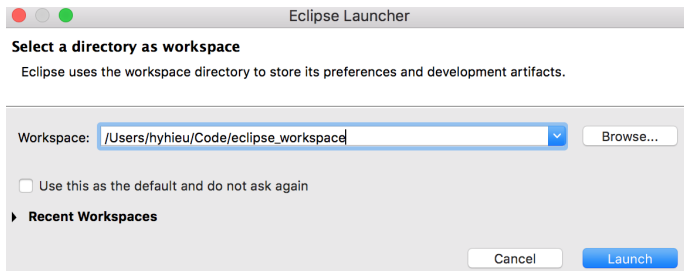
- Much less frustration when working on large projects

- 1 Install the Required Software
- 2 Getting Started with Your Project
- 3 Eclipse Basic Usages**
- 4 Navigating the Starter Code
- 5 Your Work
- 6 Java Basics

Launch Eclipse and Create your Workspace

Carnegie Mellon

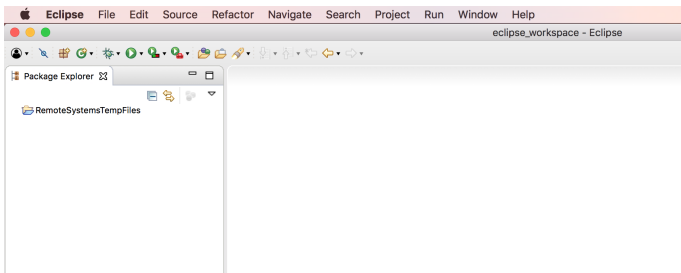
- This is **not** where your code lives



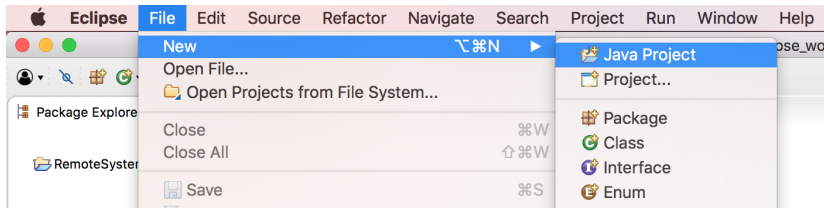
Launch Eclipse and Create your Workspace

Carnegie Mellon

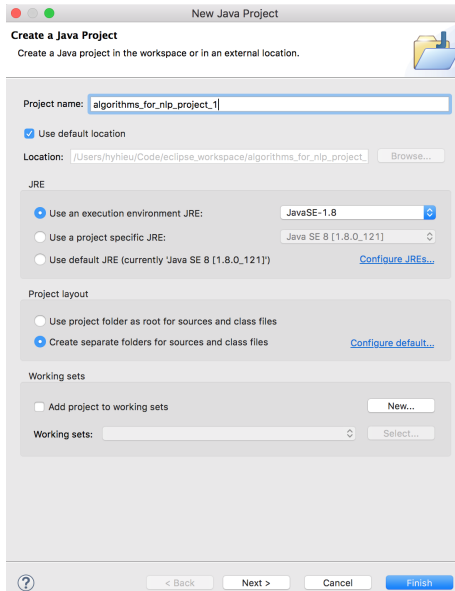
- Looks like this when done correctly



Create Your Eclipse Project



Create Your Eclipse Project



The screenshot shows the 'New Java Project' dialog box in the Eclipse IDE. The title bar reads 'New Java Project'. Below the title bar, the text 'Create a Java Project' is followed by the instruction 'Create a Java project in the workspace or in an external location.' and a folder icon. The 'Project name' field contains 'algorithms_for_nlp_project_1'. The 'Use default location' checkbox is checked. The 'Location' field shows the default path '/Users/hyhlieu/Code/eclipse_workspace/algorithms_for_nlp_project_' with a 'Browse...' button. The 'JRE' section has three radio buttons: 'Use an execution environment JRE:' (selected), 'Use a project specific JRE:', and 'Use default JRE (currently 'Java SE 8 [1.8.0_121]')'. The first option has a dropdown menu showing 'JavaSE-1.8'. The second option has a dropdown menu showing 'Java SE 8 [1.8.0_121]'. There is a 'Configure JREs...' link. The 'Project layout' section has two radio buttons: 'Use project folder as root for sources and class files' and 'Create separate folders for sources and class files' (selected). There is a 'Configure default...' link. The 'Working sets' section has a checkbox 'Add project to working sets' and a 'New...' button. Below this, the 'Working sets:' label is followed by a dropdown menu and a 'Select...' button. At the bottom, there are buttons for '< Back', 'Next >', 'Cancel', and 'Finish'. A help icon (?) is on the bottom left.

New Java Project

Create a Java Project
Create a Java project in the workspace or in an external location.

Project name:

☒ Use default location

Location:

JRE

☒ Use an execution environment JRE:

☐ Use a project specific JRE:

☐ Use default JRE (currently 'Java SE 8 [1.8.0_121]') [Configure JREs...](#)

Project layout

☐ Use project folder as root for sources and class files

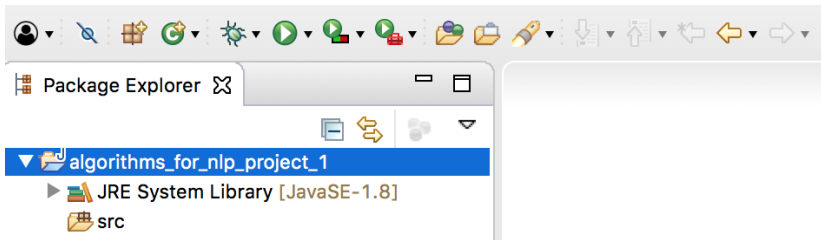
☒ Create separate folders for sources and class files [Configure default...](#)

Working sets

☐ Add project to working sets

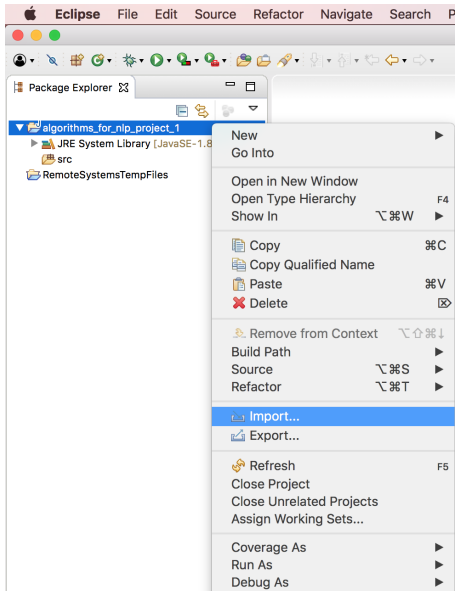
Working sets:

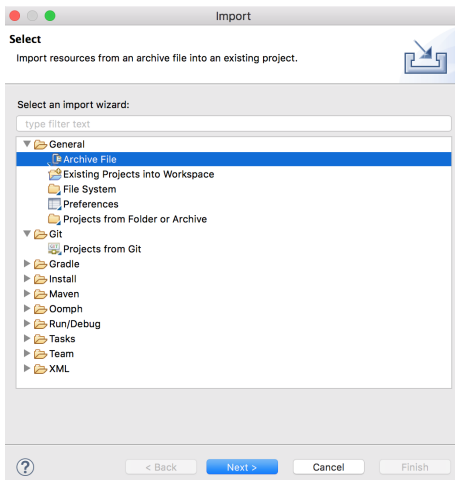
- The project folder lives in your workspace



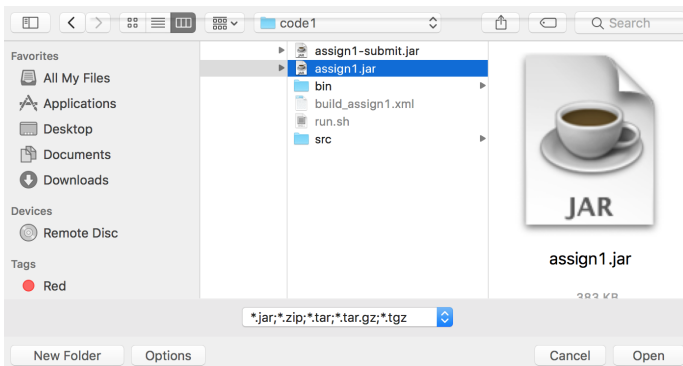
- The code **does not**.

- 1 Install the Required Software
- 2 Getting Started with Your Project
- 3 Eclipse Basic Usages
- 4 Navigating the Starter Code
- 5 Your Work
- 6 Java Basics

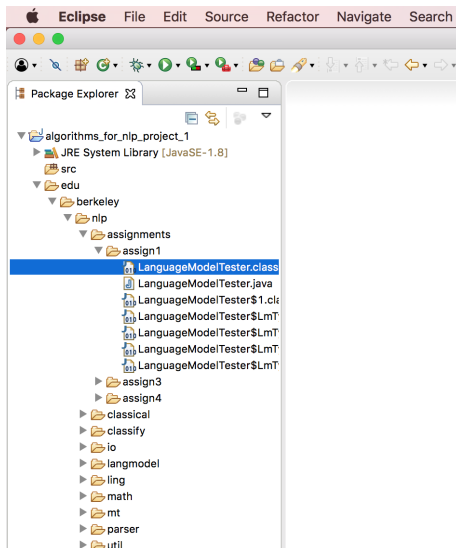




- Point to where you downloaded the code

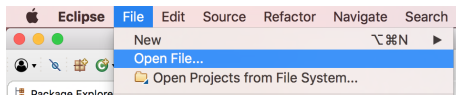


- Looks like this if done correctly

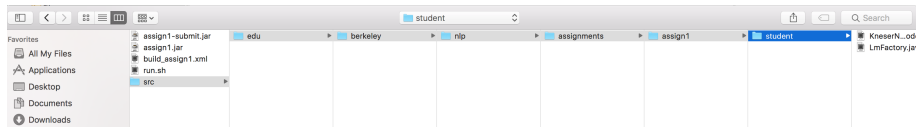


- 1 Install the Required Software
- 2 Getting Started with Your Project
- 3 Eclipse Basic Usages
- 4 Navigating the Starter Code
- 5 Your Work**
- 6 Java Basics

- Open the file `LmFactory.java`



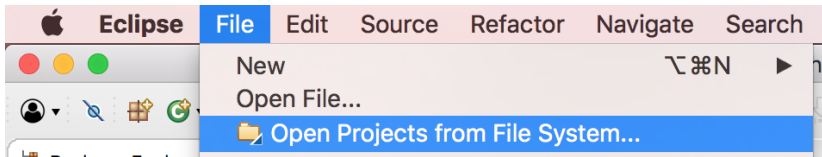
- It's tricky



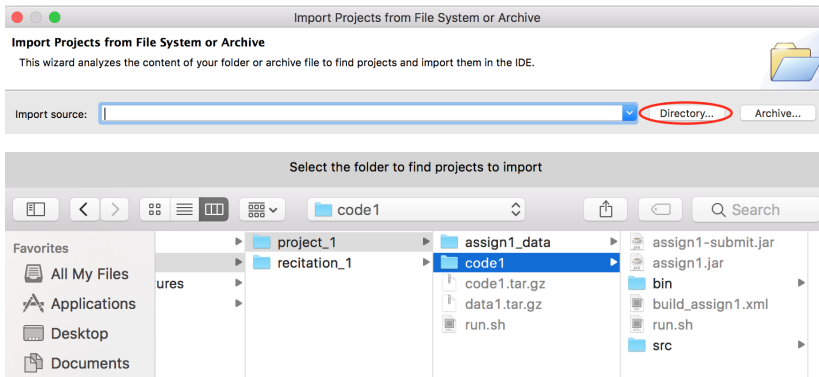
- Looks like this if done correctly

```
LmFactory.java
1 package edu.berkeley.nlp.assignments.assign1.student;
2
3 import java.util.List;
4
5 import edu.berkeley.nlp.langmodel.LanguageModelFactory;
6 import edu.berkeley.nlp.langmodel.NgramLanguageModel;
7 import edu.berkeley.nlp.assignments.assign1.student.KneserNeyLanguageModel;
8
9 public class LmFactory implements LanguageModelFactory
10 {
11
12     /**
13      * Returns a new NgramLanguageModel; this should be an instance of a class that you implement.
14      * Please see edu.berkeley.nlp.langmodel.NgramLanguageModel for the interface specification.
15      *
16      * @param trainingData
17      */
18     public NgramLanguageModel newLanguageModel(Iterable<List<String>> trainingData) {
19         return new KneserNeyLanguageModel(trainingData);
20     }
21
22 }
```

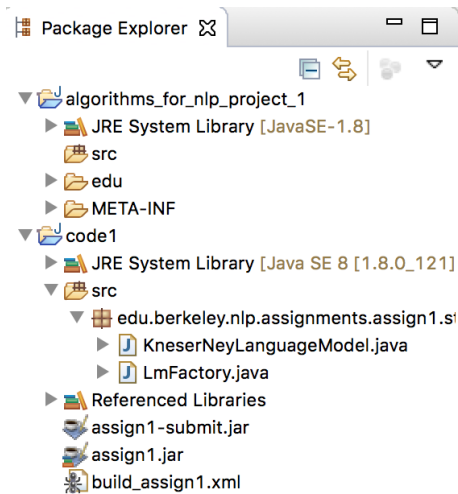

- Easier to navigate and organize your files



- Select the folder with `build_assign1.xml`



- Looks like this if done correctly



- ① Install the Required Software
- ② Getting Started with Your Project
- ③ Eclipse Basic Usages
- ④ Navigating the Starter Code
- ⑤ Your Work
- ⑥ Java Basics

- 15-110: Principles of Computing
- <http://www.cs.cmu.edu/~jxc/100.html>

Questions?