Introduction to Java

Michelle Brush
Senior Software Architect
Cerner Corporation

Class Structure

- The Why and What of Java

- Basic Logic

- Object-Oriented Programming

- When Things Go Wrong

- **Common Java APIs**

- JUnit

- Generics

- Putting It All Together I & II

Why is Java popular?

Community

Support
Libraries
Best Practices
Resources
Conferences
Tools

What's a library?

In Java, it's a collection of **classes** and **interfaces that can be used** by other programmers.

Application **P**rogramming **I**nterface

A **specification** for how software can interact to accomplish some work.

In Java, a collection of **interfaces** and **classes** that allow you to **reuse** someone else's **code**.

You'll have to use libraries to do anything interesting.

Like...

call a web service

write to a database

read or write from a file

create a user interface

How do you use a library?

#1
It needs to either be in the **JDK** or in the **classpath**.

JDK: Java Development Kit (the built-in stuff)

Classpath: Locations (paths or folders) you tell Java to look for **jars**.

JAR: Java ARchive File

A bunch of .class files.

A zip file of compiled classes and interfaces packaged together.

What's a class file again?

It's Java code compiled into bytecode.

To use someone else's library you need their class files - packaged up into a **jar**.
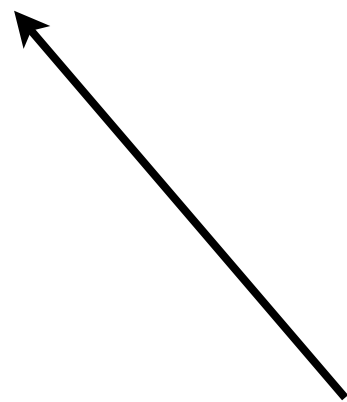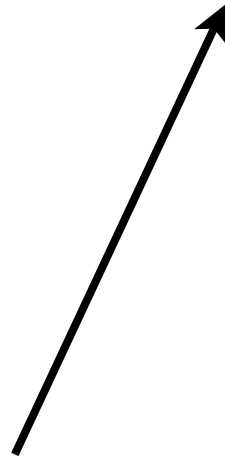
#2
You have to **import** it.

import `java.util.List;`

Import command

```
import java.util.List;
```

package

```
import java.util.List;
```
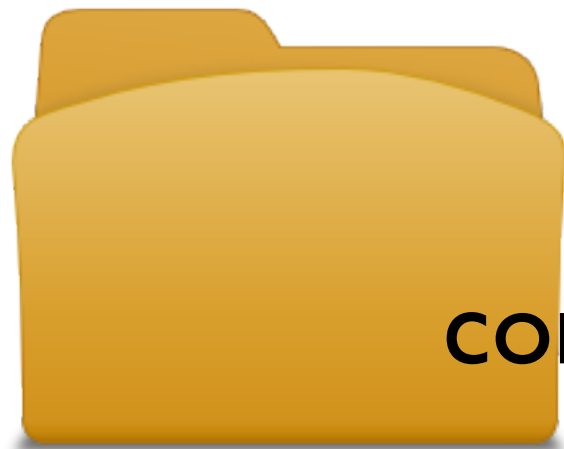
class or interface

Package: A way to organize classes and interfaces into containers

Creates **scope** for a name

A package is a folder.

Each dot indicates a nested folder.

com

girldevelopit
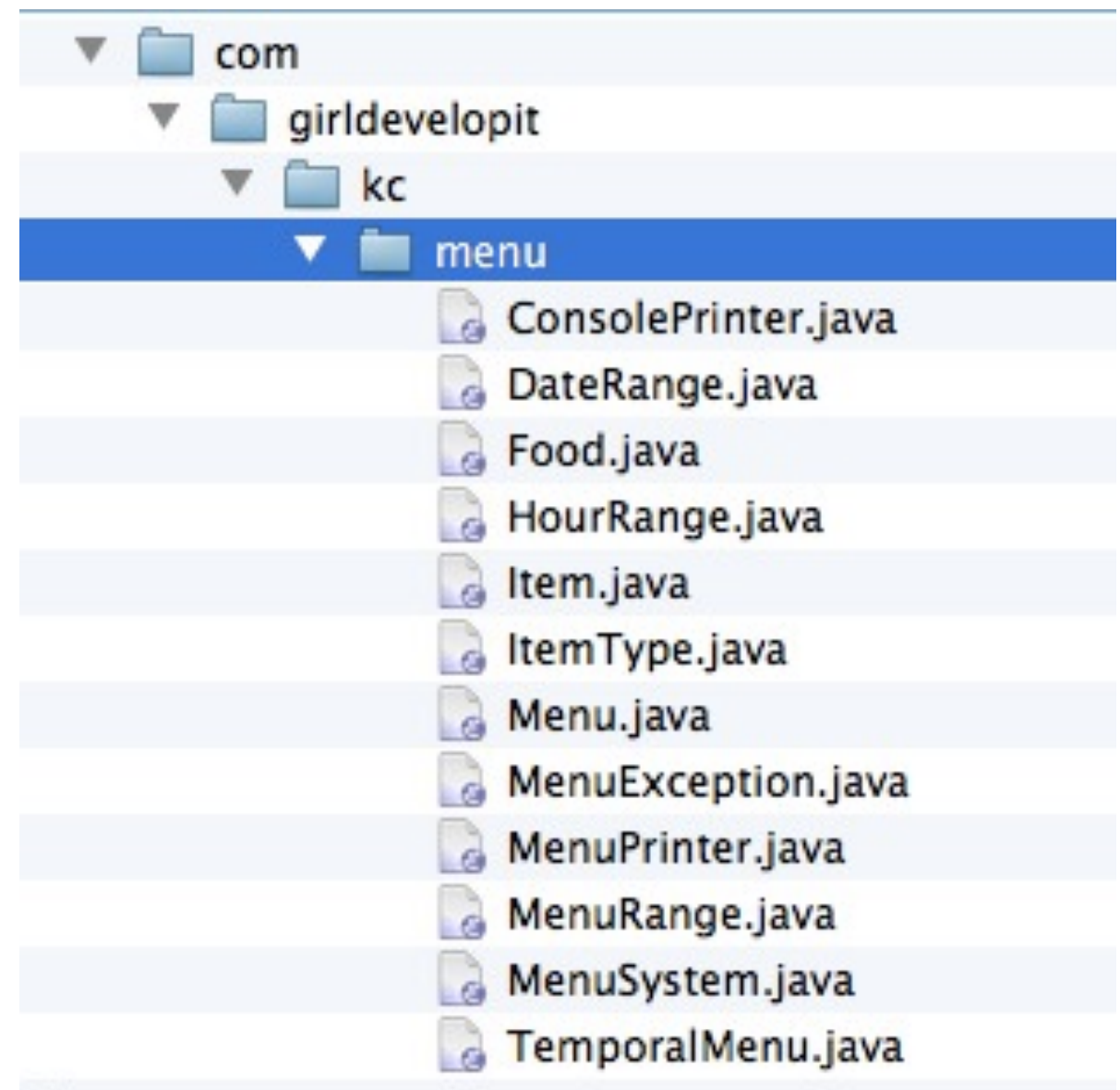
com.girldevelopit.kc

kc

Declare the package at the top of the Java file:

```
package com.girldevelopit.kc;
```

```
package com.girldevelopit.kc.menu;
```

If you declare a package, the .java file has to be in that folder.

```
▼ 📁 com
    ▼ 📁 girldevelopit
        ▼ 📁 kc
            ▼ 📁 menu
                📄 ConsolePrinter.java
                📄 DateRange.java
                📄 Food.java
                📄 HourRange.java
                📄 Item.java
                📄 ItemType.java
                📄 Menu.java
                📄 MenuException.java
                📄 MenuPrinter.java
                📄 MenuRange.java
                📄 MenuSystem.java
                📄 TemporalMenu.java
```

You have to know what **package** something is in to **import** it.

```
import java.util.Calendar;
```

You have to import it in **each file** you want to use its name.

#3
You have to read the **JavaDoc**.

JavaDoc are special comments.

```
/**
 *
 */
```

The special comments are extracted.

```
/**
 *
 */
```

They describe how to use the code.

```
/**
 *
 */
```

Like what the function does.

```
/**
 *
 */
```

What parameters it expects

```
/**
 *
 */
```

What exceptions it throws

```
/**
 *
 */
```

Let's browse some JavaDoc.

http://docs.oracle.com/javase/7/docs/api/java/util/List.html

Useful APIs in the JDK

Collections
Math
Swing
I/O

Other Useful APIs

JUnit
Apache Commons
Jersey
JodaTime

We're going to use the the **java.util.io** package to write the menus to files.