**CSC 133**
**Object-Oriented Computer Graphics Programming**

# Introduction to Codename One
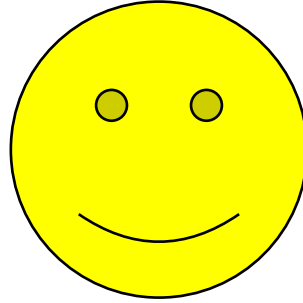
Dr. Kin Chung Kwan

*Spring 2023*

Computer Science Department
California State University, Sacramento

SACRAMENTO STATE

# **Assignment 0 is released!**

## Enjoy

# Course Description

## csc 133 - Obj-Oriented Cmptr Graph - 3 Units

Introduction to computer graphics and to advanced topics in object-oriented programming. Mobile application development; implementation of event-driven systems; advanced object-oriented concepts such as inheritance and polymorphism; implementation of software design patterns; graphical user interface development; fundamentals of 2D graphics systems. Application of these topics to mobile programming. Prerequisite: CSC 130, CSC 131, not currently enrolled in CSC 133.

*Click on the first column in each table row to see class details.*

# Mobile

**> 80%** people have a smartphone (or more)
- iPhone, Android, Window Phone, etc.

We use it everyday
- Information retrieval
- Communication
- Security
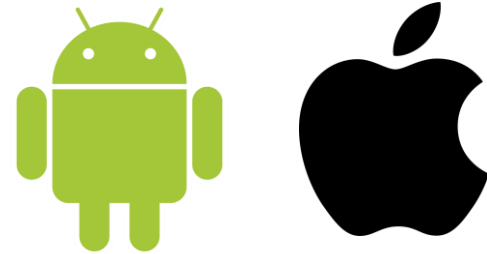- Entertainment: Video / Gaming!

# Why Mobile?

- Growing need for App developers
- Make it fun and cool! ☺
- Contemporary
- A base for further exploration
- Stronger **resume**

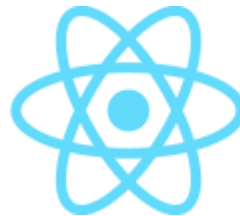# Mobile App Development

Platform specific
- Android studio, XCode

Web-based
- HTML5

Native cross-platform
- Codename One, React Native, Xamarin

# Codename One (CN1)

- Released in 2012
- Open-Source
- **Free**!
- Java-based
- Cross-platform
- With simulator

# CN1 vs Java

Subset of Java

- (Initial) subset of Java 1.3
- (Then) support subset of Java 5, 8, and 11

Does not support Java features that are not suitable for mobile devices. E.g.

- Reflections
- Desktop APIs such as java.net, java.io.File etc. (provides its own alternatives)
- Swing library (provides Swing redesigned for mobile environment in its UI API/package)

# CN1 Builder

Provides drag and drop tools to create GUI components

We will not use it
- We are programmer
- Want to learn more

# Download & Installation

# Java Development Kit

- Download and Install **JDK 11**
    - https://www.oracle.com/java/technologies/downloads/#java11
- You may need to register an Oracle account

# IDE Installation

For CSC133, we require ⬤ **eclipse**® for Java

**Eclipse IDE for Java Developers 2022-03**

Download:

- https://www.eclipse.org/downloads/packages/release/2022-03/r

Eclipse IDE 2022-03 R Packages

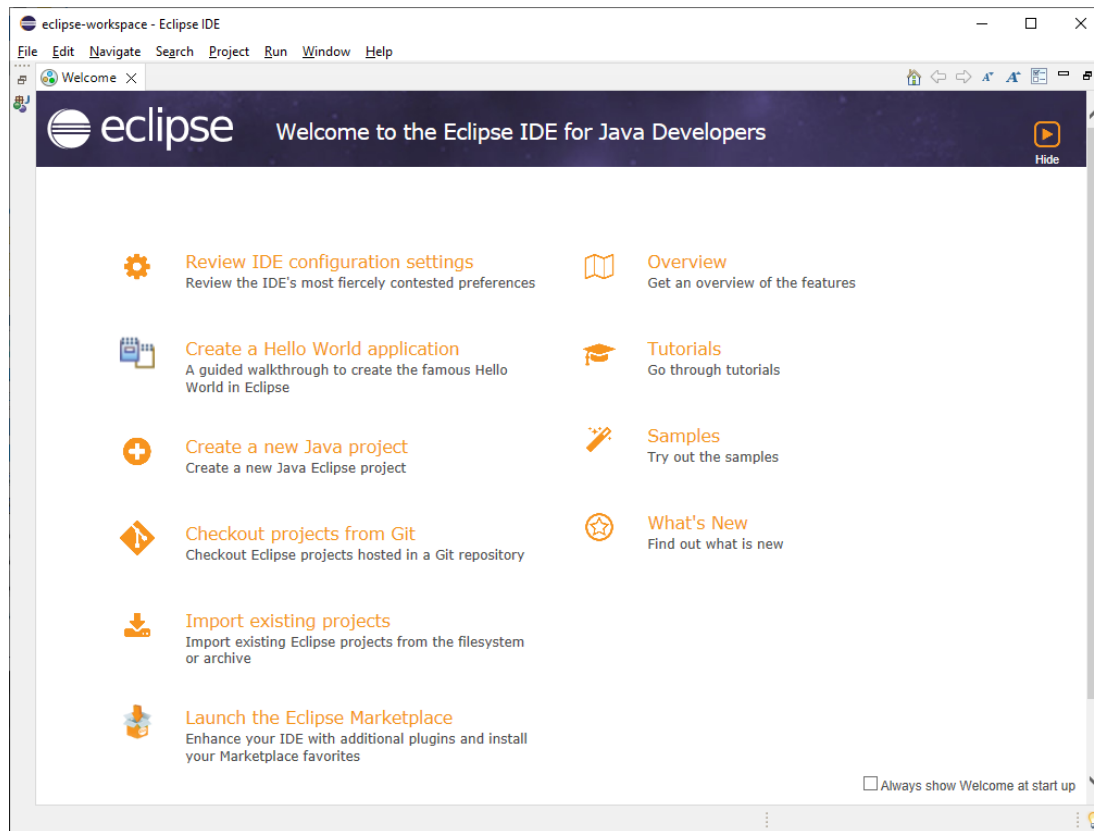Eclipse IDE for Java Developers

308 MB    1,226,151 DOWNLOADS

The essential tools for any Java developer, including a Java IDE, a Git client, XML Editor, Maven and Gradle integration

Windows x86_64
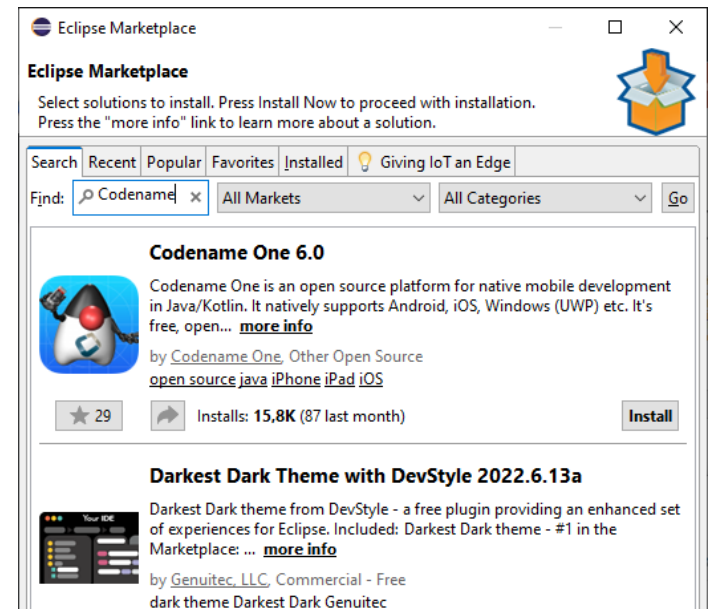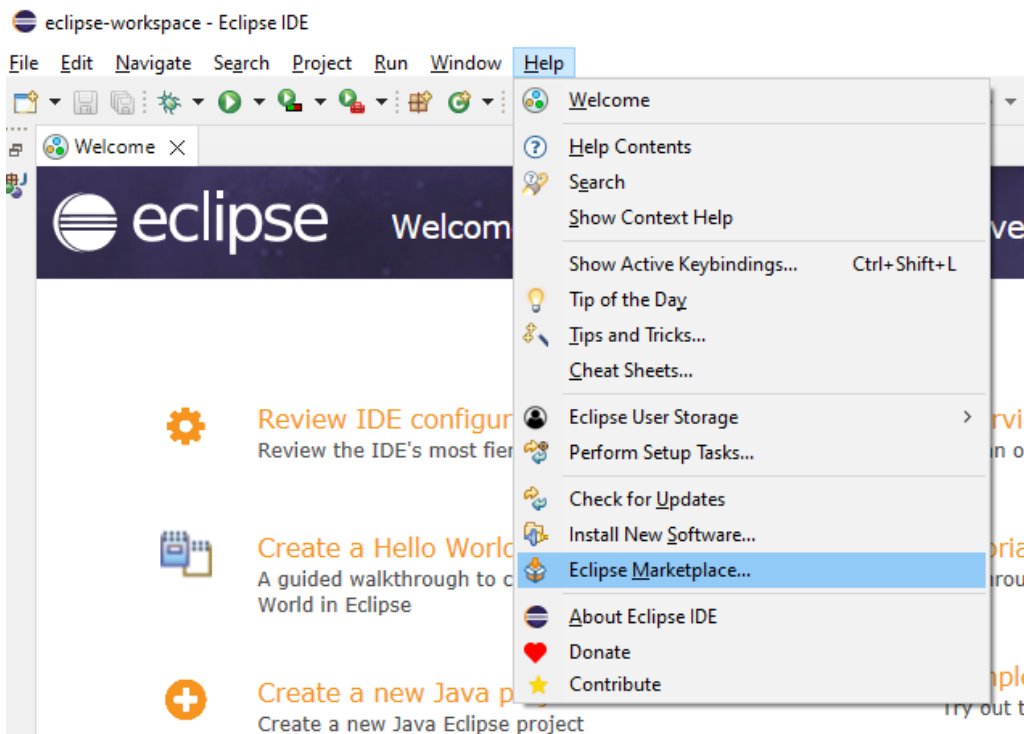macOS x86_64 | AArch64
Linux x86_64 | AArch64

# Run the IDE

- Eclipse 2022-03 (4.23)

# CN1 Installation

"Help" > "Eclipse Marketplace"

Search for "**Codename One 6.0**" and install

# Versions*

| | |
|---|---|
| Java SE JDK | **11** |
| Eclipse IDE for Java Developers | **2022-03R (v 4.23)** |
| Codename One | **6.0** |
| Operation System | **Windows** |

*Current version in our lab and for testing.

You can use another version **at your own risk**.

# Alternative

If no "Eclipse Marketplace"

- Select "Help" → "Install New Software"

- Click the "Add" button on the right side

- Name = Codename One

- Location = https://codenameone.com/files/eclipse/site.xml

- Select the entries & follow the wizard to install

# **Notes**

Detailed instructions

https://www.codenameone.com/how_di_i/how-do-i-create-a-basic-hello-world-application-send-it-to-my-device-using-eclipse.html
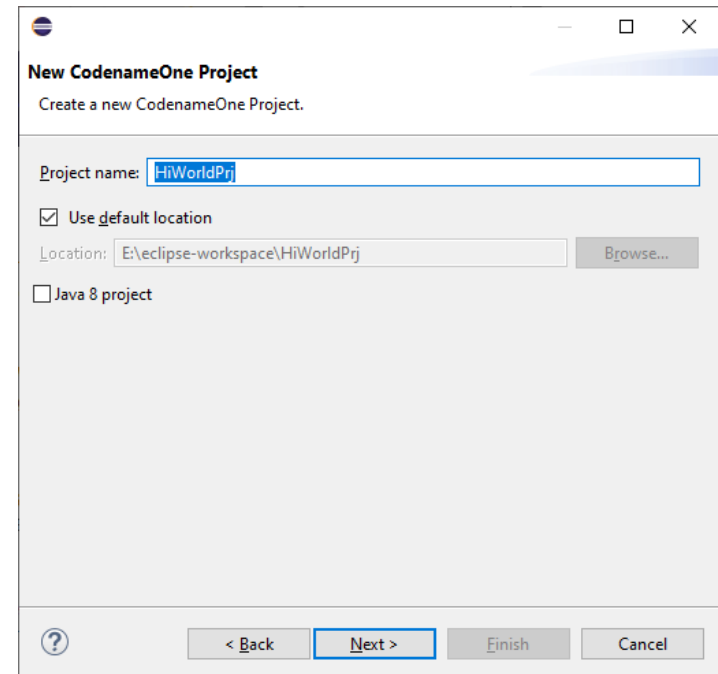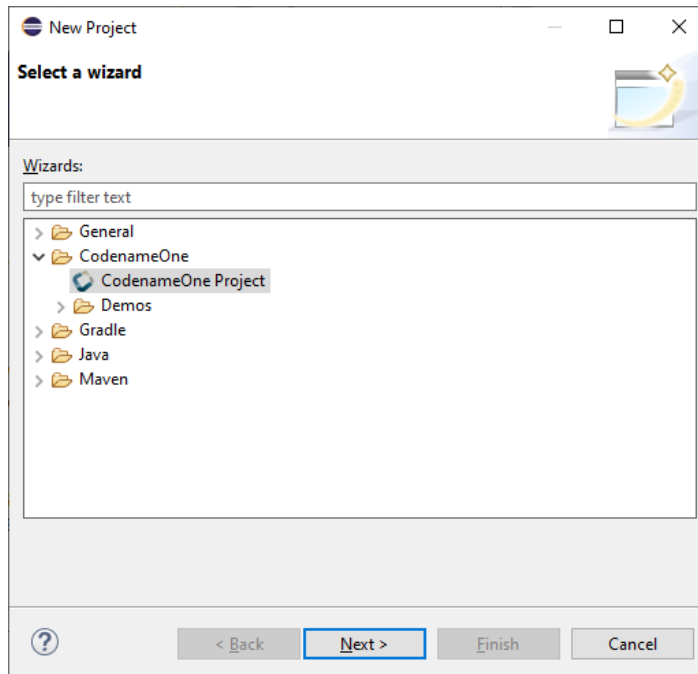
They are installed at school machines.

- Listed in the outline

# Ready to Code!

Let's try "**Hello, world**!"

# Create a Java Project

- Create a new eclipse project
    - File > New > Project > Codename One Project
    - Give a project name "HiWorldPrj." Hit "Next".
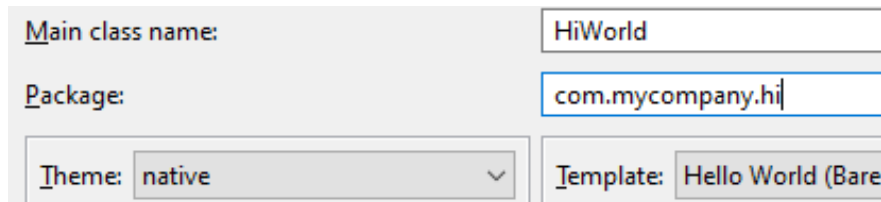
# Setup the Project

Given a main class name "**HiWorld**"

- name of your first class

Package name "**com.mycompany.hi**"

- Unique identification

Select

- "native" for theme
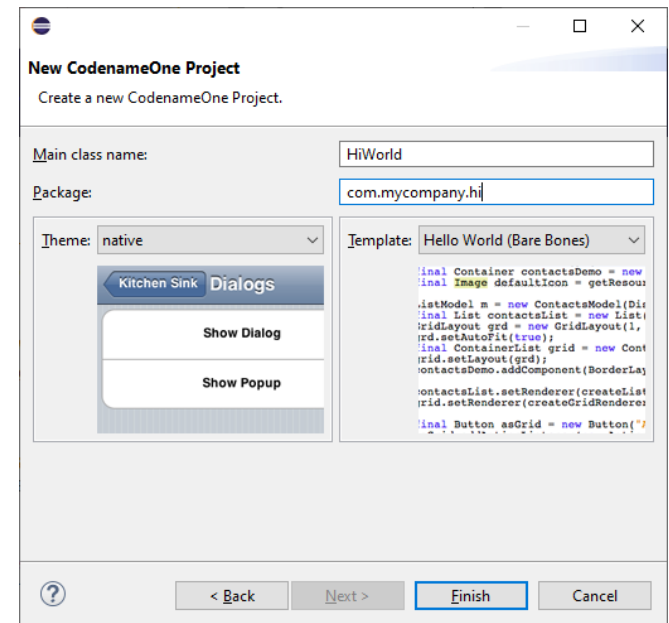- "Hello World (Bare Bones)" for Template

**Created!**

**Close the welcome page**
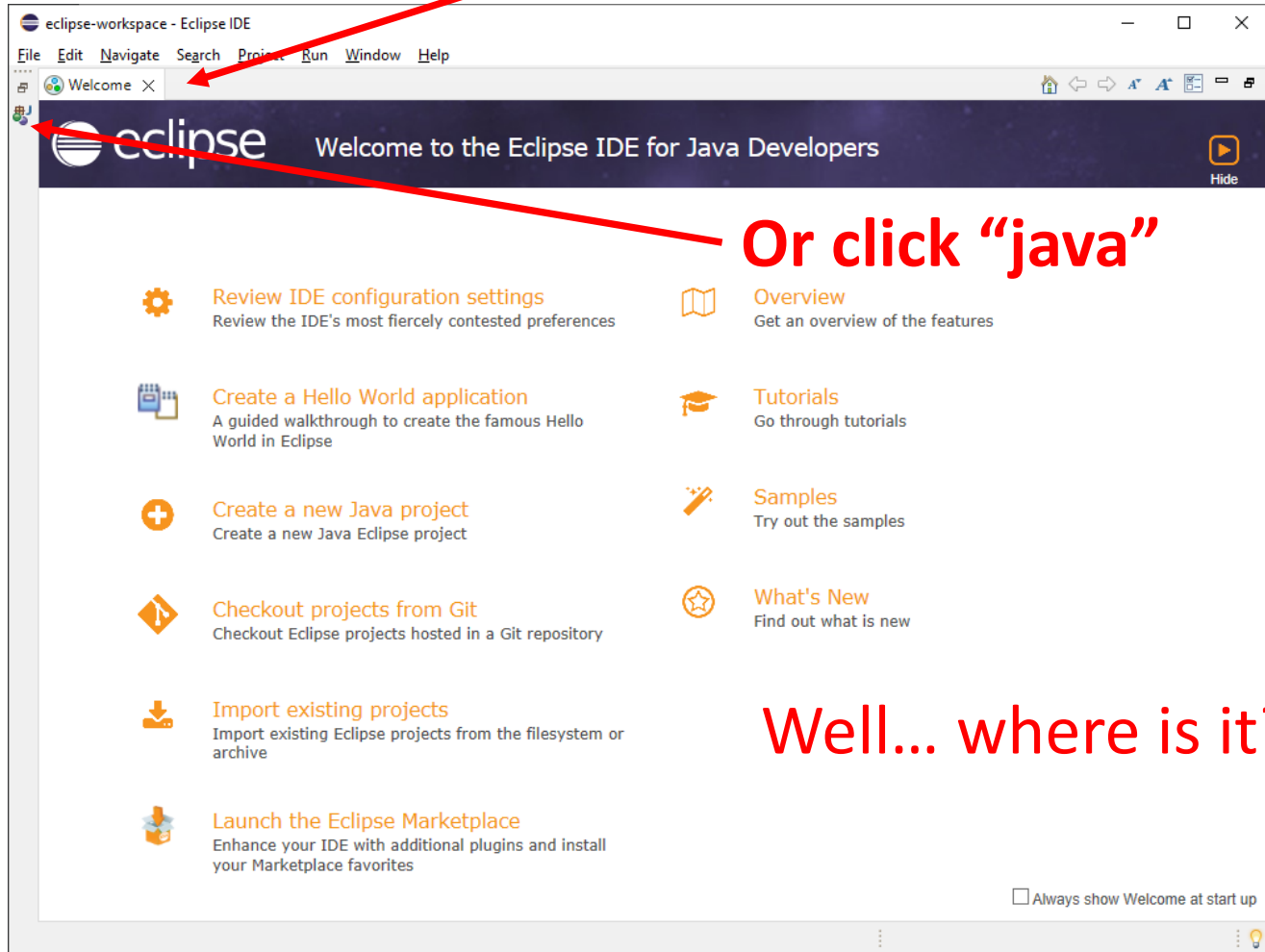


**Or click "java"**

**Well… where is it?**

# Your Files

Your main class is in package explorer

- HiWorldPrj > src > com.mycompany.hi > HiWorld.java
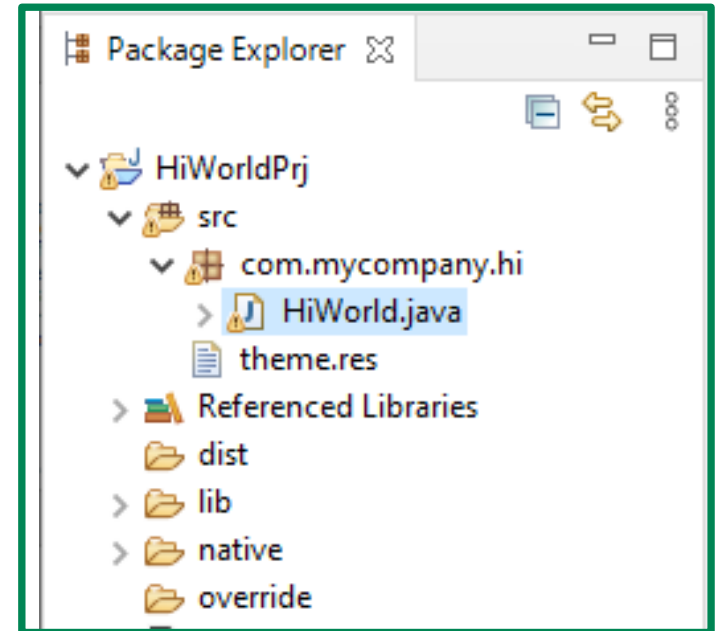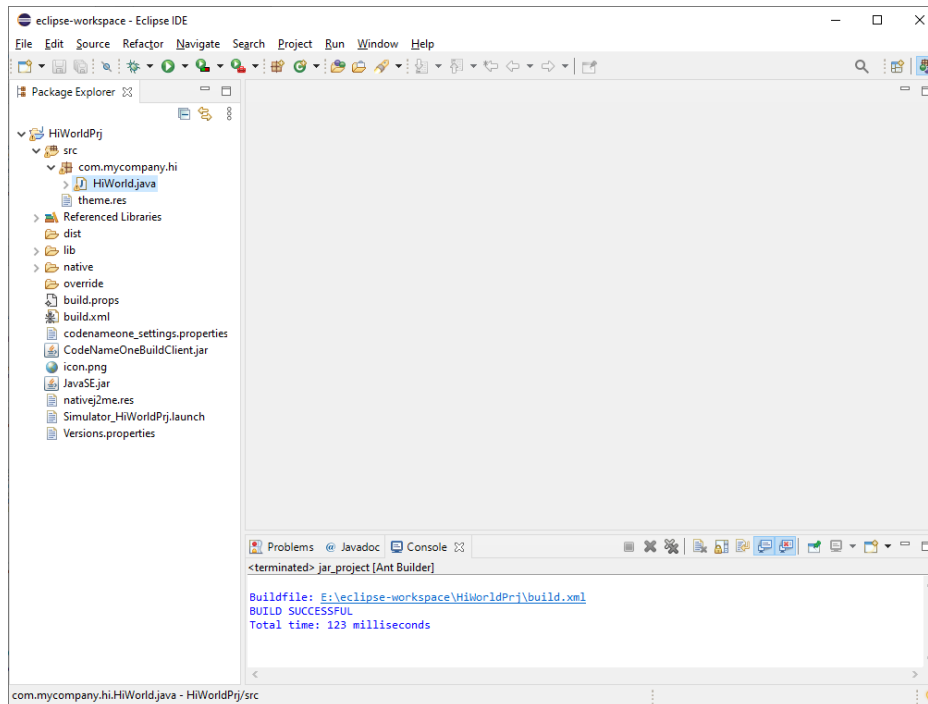
# Execute the Java

Right click "Simulator_HiWorldPrj.launch"
> Run as / Debug as

Or press F11 if you selected the file
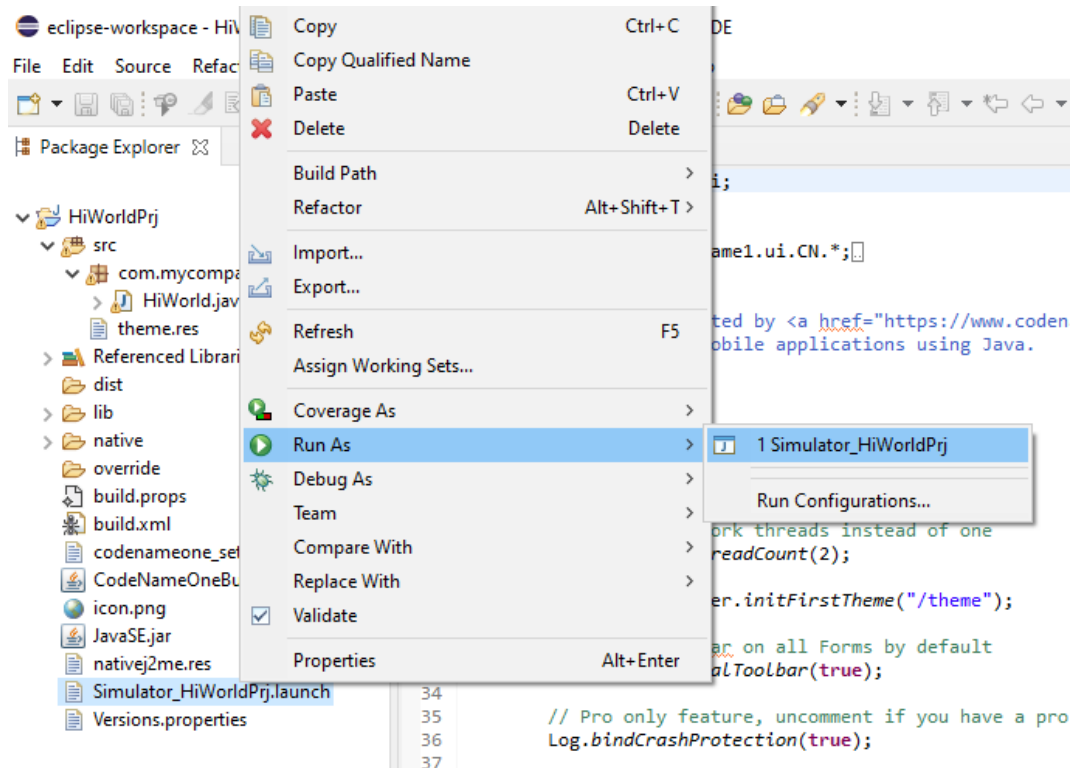
# By command line

**Windows**:
*java -cp dist\HiWorldPrj.jar;JavaSE.jar com.codename1.impl.javase.Simulator com.mycompany.hi.HiWorld*

**Unix-like operating systems:**
*java -cp dist/HiWorldPrj.jar:JavaSE.jar com.codename1.impl.javase.Simulator com.mycompany.hi.HiWorld*

*( Note: use / and : instead of \ and ; )*

All in one line, but put spaces between sub-lines

# Simulator

"hi World"
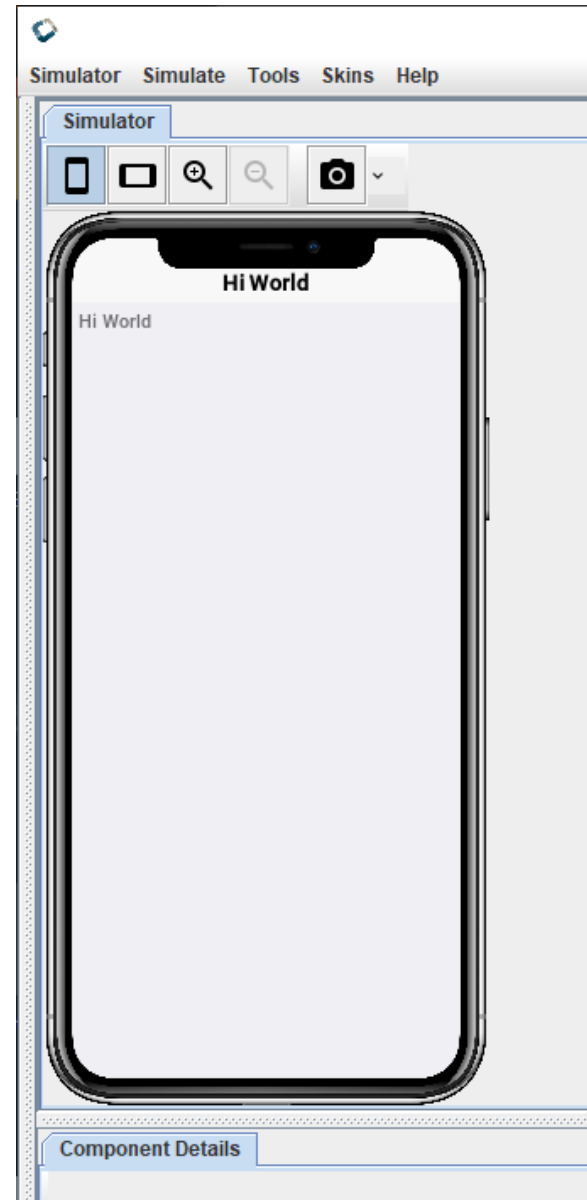
# iPad III

Change the skin to different smartphone:

- Use ipad3_os7.skin

- Skin > More > iPad III

- Use in assignments

# **About the Assignment**

Assignment instructions

# Assignments

- For each assignment create a different CN1 project.

- You **must** create all assignments in the same way as HiWorldPrj example:
  - uncheck "Java 8 project", select "native" theme, and **"Hello World (Bare Bones)"** template.
  - change the project, main class, and package names…

# Assignments Naming

- Project Name:
  - **A#Prj**
  - (# is the project number)
  - E.g., A1Prj, A2Prj
- Main Class Name:
  - **Starter** (For all assignments)
- Package: **com.mycompany.a#**

# Main Class Structure

*public class Starter {*

    *…*
    *public void init( … ) { … }*
    *public void start( ) { … }*
    *public void stop( ) { … }*
    *public void destroy( ) { }*
*}*

Eclipse will generate them.

# How to Start?

- Modifying **start()** in *Starter.java*

- Do **NOT** delete other methods

- Adding more java files
  - right click the package
  - New > Class
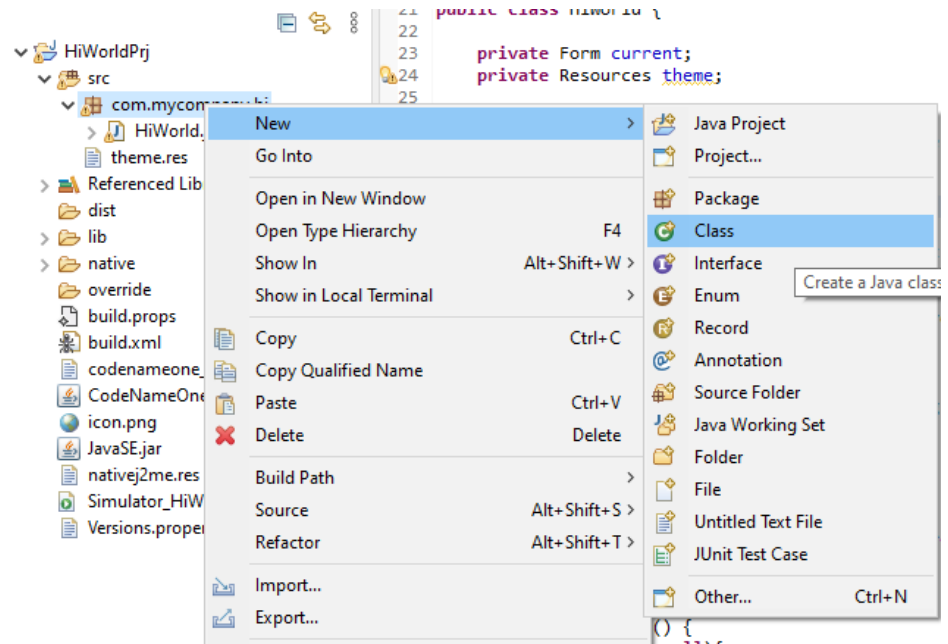
# Make Sure

**dist\A1Prj.jar is up to date!**

- If not, in Eclipse, right click on "dist," "build" or project directory and hit "Refresh."
- Or press F5

**Your submission works from command-line**

Go into the A1Prj directory and type:

*java -cp dist\A1Prj.jar;JavaSE.jar*
*com.codename1.impl.javase.Simulator*
*com.mycompany.a1.Starter*
*(Notes: all in one line, but put spaces between sub-lines)*

# Example - Cause

KC worked on his assignment until 11:29pm and submitted it. Because of some **mysteries**, CN1 did not update the .jar file. His latest code are not included in the .jar file and thus failed the assignment.

# Solution
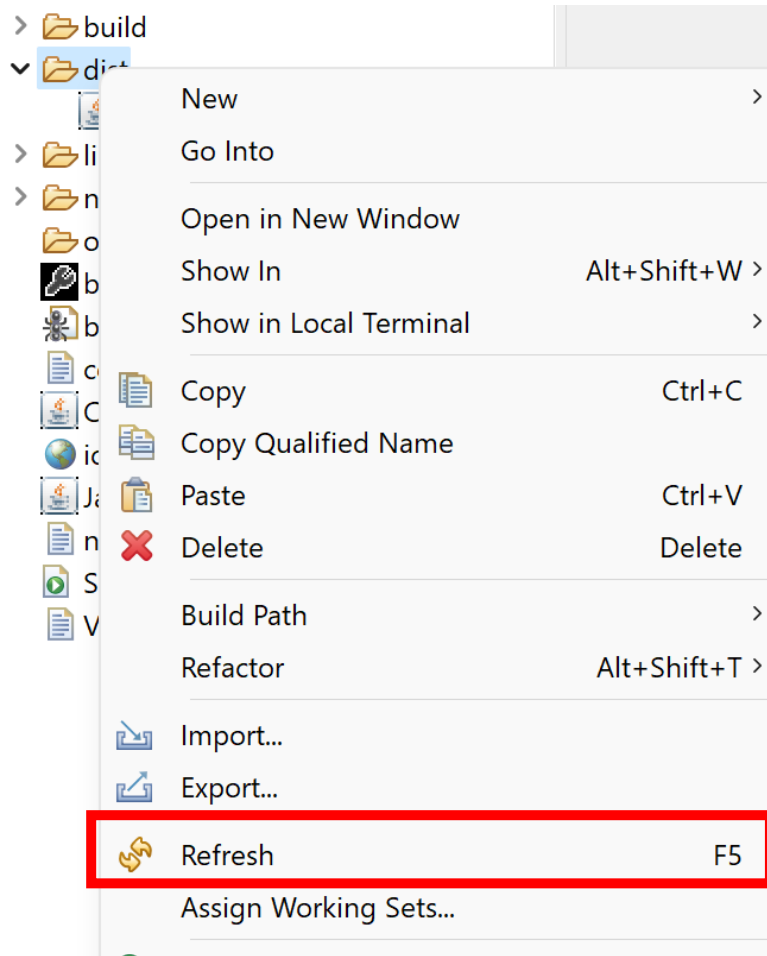


KC retakes CSC133, now he refreshes it!



The .jar is up-to-date!

# Command-Line

- To enter command-line in eclipse
    - Right click project > show in local terminal > terminal
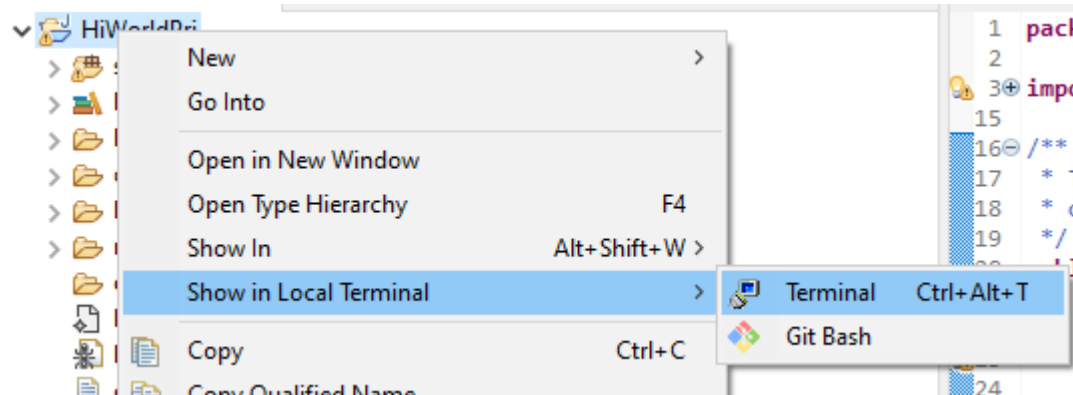


    - The terminal appear at the bottom
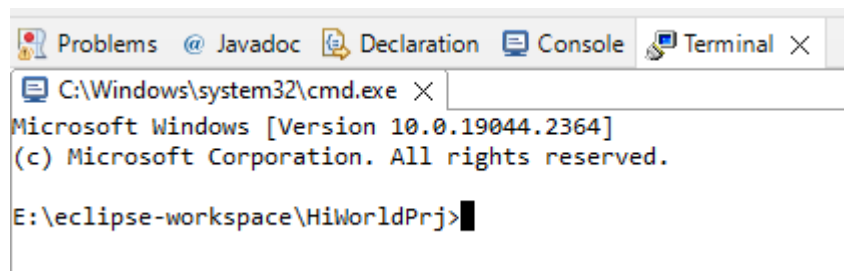
# Submission

For ALL assignments, submit **TWO** files:

1. Kin-Chung-Kwan-a1.zip (**ZIP** file)

   *(Note: Your name and the assignment index. Use hyphen to replace white space.)*

2. readme.txt (**TEXT** file, not pdf or doc)



Kin-Chung-Kwan-a1.zip

readme.txt

# **Zip File**

Kin-Chung-Kwan-a1.zip

## Structure of the ZIP file

```
Kin-Chung-Kwan-a1.zip
 ├→ A1Prj.jar
 ├→ A1UML.pdf
 └→ src
        ├→ theme.res
        └→ com
               └→ mycompany
                      └→ a1
                            ├→ starter.java
                            ├→ xxxx.java
                            └→ …
```
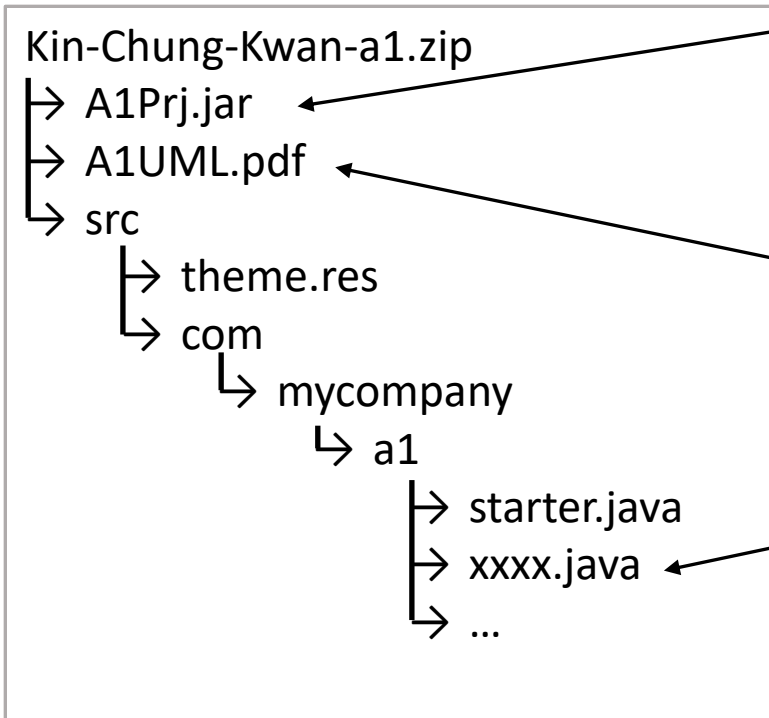
The *.jar* file generated under dist directory

Some assignments require you to provide UML diagram

Include all your *.java* files

# Text file


readme.txt

- Test your *.jar* and codes in a lab machine
- Indicates the remotely accessible lab machine
    - That generate/test your submitted *.jar* files and code.
    - By the **lab number** and the **name of the machine**
- Any additional information you want to share
- You may receive the grader comments on your text file

Do NOT place this TEXT file inside the ZIP file, instead submit it separately.

# Ready for Assignment?

Assignment 0

# Assignment#0

- Find a lab computer that has CN1
  or install CN1 to your computer.

- Following the instruction in this slides

- Generate an empty project called **A0Prj**.

- Modify *Starter.java*
  - **replacing the texts "Hi World" with "Assignment#0"**

- Experiments with the IDE.

- Verify that your submission also works from the command line.

# Assignment#0 (cont.)

Verify that your program works on a lab machine,
- Such as a machine in a **<u>remotely accessible</u>** lab.

To build and test your assignment in the lab
- Copy your assignment directory located in the Eclipse workspace directory of your machine to the Eclipse workspace directory of the lab machine
- File > Import > General > Existing Project into Workspace

# Deadline

# No deadline!

1. You don't need to submit it!
2. Do <u>NOT</u> submit it via Canvas!
3. No submission system for it

## Not one of the four programing assignments

# Troubleshooting Problem I

- If dist\HiWorldPrj.jar is not generated:
    - Signup for a CN1 account at https://www.codenameone.com
    - Right click on project and hit "Codename One → Send Android Build"
    - hit OK if a warning is given, then login with your CN1 account

- If still does not work, set JDK environment variable:
    - Goto "Control Panel > System > Advance System Settings > Environment Variables"
    - Add JAVA_HOME as "C:\Program Files\Java\jdk-11" to "System Variables"

# Troubleshooting Problem II

If the command line complains that:

- 'java' is not recognized

  - add "C:\Program Files\Java\jdk-11\bin" to PATH

- JavaSE.jar cannot be found :

  - Make sure you are in the project directory that has JavaSE.jar

  - Add current directory (indicated by a single period ".") to CLASSPATH

# Troubleshooting Problems III

Make sure you are using JDK 11 to run:

- Eclipse IDE itself:
  - set JAVA_HOME to C:\Program Files\Java\jdk-11 o
  - If this does not work, edit the "eclipse.ini" file to force Eclipse to use your JDK 8 install. See https://wiki.eclipse.org/Eclipse.ini for more info.

- Applications from command-line:
  - add C:\Program Files\Java\jdk-11\bin to PATH

Make sure "compiler compliance level" in Eclipse is set to 1.8:

- Check from "Window > Preferences > Java > Compiler" (also check project specific preferences and make sure this level is set to 1.8)

# CN1 Online Resources

- Developers guide

CN1 Developer Guide - Revision 3.6 (pdf is available at Canvas)

- Video tutorials

https://www.codenameone.com/how-do-i.html

(note: mostly give examples that use the GUI builder which we will not utilize)

- JavaDocs for APIs

https://www.codenameone.com/javadoc/index.html

# Any Questions?

# Free to Go!