**Week 8: 18 October**

**Session**

* announcements
  + any remaining questions about project 3
  + reminders: Quiz 2 next week (2 SE Radio podcasts), test 2 the following week
* principles of object-oriented programming: SOLID
  + S - Single Responsibility Principle
  + O - Open Closed Principle
  + L - Liskov Substitution Principle
  + I - Interface Segregation Principle
  + D - Dependency Inversion Principle
  + References
    - [Uncle Bob's Principles of OOD](http://butunclebob.com/ArticleS.UncleBob.PrinciplesOfOod)
    - [Pablo's SOLID Software e-book](http://lostechies.com/wp-content/uploads/2011/03/pablos_solid_ebook.pdf)
    - [SOLID Principles in C#](http://www.codeguru.com/columns/experts/solid-principles-in-c-an-overview.htm)
* SOLID and other basic object-oriented design principles ("SOLID + 2"): [presentation](http://www.slideshare.net/intellizhang/the-oo-design-principles)
  + [overview](http://en.wikipedia.org/wiki/SOLID_(object-oriented_design))
  + [extended overview by Uncle Bob](http://butunclebob.com/ArticleS.UncleBob.PrinciplesOfOod) with links to detailed articles
  + [information hiding/minimize coupling](http://c2.com/cgi/wiki?InformationHiding)/Law of Demeter ([Tarr](http://userpages.umbc.edu/~tarr/dp/lectures/OOPrinciples.pdf) p1-)
  + [favor composition over inheritance](http://c2.com/cgi/wiki?CompositionInsteadOfInheritance) ([Tarr](http://userpages.umbc.edu/~tarr/dp/lectures/OOPrinciples.pdf) p9-)
    - Coad's rules (Tarr p22-)
    - role-based design (Tar p23-)
  + dependency inversion principle/design with interfaces ([Tarr](http://userpages.umbc.edu/~tarr/dp/lectures/OOPrinciples.pdf) p33-, [Coad](http://www.petercoad.com/download/bookpdfs/javadesign2ndeditionchapter3.pdf))
  + open-closed principle ([Tarr](http://userpages.umbc.edu/~tarr/dp/lectures/OOPrinciples.pdf) p40-)
  + Liskov substitution principle ([Tarr](http://userpages.umbc.edu/~tarr/dp/lectures/OOPrinciples.pdf) p51-)
  + [single-responsibility principle](https://docs.google.com/open?id=0ByOwmqah_nuGNHEtcU5OekdDMkk) (cohesion part 1)
  + [interface segregation principle](http://docs.google.com/a/cleancoder.com/viewer?a=v&pid=explorer&chrome=true&srcid=0BwhCYaYDn8EgOTViYjJhYzMtMzYxMC00MzFjLWJjMzYtOGJiMDc5N2JkYmJi&hl=en) (cohesion part 2)
  + package-level principles: cohesion and coupling
    - acyclic dependencies
* Android example programs
  + Android framework
    - [architecture](http://www.techotopia.com/index.php/An_Overview_of_the_Android_Architecture)
    - [overview](http://developer.android.com/guide/components/fundamentals.html)
    - [activities and their lifecycle](http://developer.android.com/guide/components/activities.html) (scroll about 60% down)
    - [tutorials](http://developer.android.com/training/)
  + [examples](https://github.com/LoyolaChicagoCode) - search for "android-java"
  + [hello-android-java](https://github.com/LoyolaChicagoCode/hello-android-java) - notification
    - HAXM (see recent post)
    - creation of AVD
    - roles of hg and Gradle
  + [simplebatch-android-java](https://github.com/LoyolaChicagoCode/simplebatch-android-java) - scrollable text output
    - functionality: scrollable
    - Android framework and [activity life cycle](http://developer.android.com/guide/components/activities.html)
    - preview of agile process
  + [simpledraw-android-java](https://github.com/LoyolaChicagoCode/simpledraw-android-java) - drawing simple shapes based on lines

**Reading/Podcasts**

* + Android 4 App Development Essentials, Chapters 5-8
  + [SE Radio episode 46 on refactoring](http://www.se-radio.net/episode-46-refactoring-pt-1/) **- will be on Quiz 2**

**Homework**

**Project 3 due Friday, October 21**