CMPUT 301
Lab2 Presentation

Lab 2 Overview

OOP Review

Android Basics

Assignment 0 Review (demo)

ListView Example and Exercise (demo)

OOP Review

- Inheritance
  - Superclass and Subclass
- Polymorphism
  - @Override
- Encapsulation
  - Why do we need encapsulation?

Inheritance

- subclass (child) - the class that inherits from another class
- superclass (parent) - the class being inherited from
- The subclass will inherit all fields and methods. (basic constructor, attributes and all the user defined functions)

Inheritance

- What does super() or super(variables) do?
  Construct its super class.
- Which constructor the program runs first? (super or sub)
  Super! Because ImportantTweet is a Tweet, so we need to be a Tweet first
- Without super() in subclass, super() would automatically called.

  (However, when your superclass doesn't have a no-arg constructor, the compiler will require you to call super with the appropriate arguments.)

Polymorphism

```
public boolean equals(Person other) {
    ...
}
```

```
@Override public boolean equals(Person other) {
    ...
}
```

Why do we use @override?

Because there is already a definition in the superclass, we cannot define a same function twice.

Encapsulation

When do I use private (or protected) variable?

Whenever you want to hide variable from another class. Whenever that data should not be known or should not be accessible by other class..

How could getters and setters hide information?

You could add restrictions before just setting or getting the data. (For example, before set the new password, require the old password first)

ListView Example and Exercise (demo)