Intents and Parcelables
How hard was the week 10 code review assignment?

A) Easy
B) Moderate
C) Challenging
D) Unreasonable
How long did week 10 assignment take?

A) Less than 2 hours
B) 2 to 4 hours
C) 4 to 6 hours
D) 6 to 8 hours
E) More than 8 hours
Assignment for Next Wednesday night

- Add a ‘Detail Activity’ to your Zomato App
  - I’ll do an example today
- Short Proposal for Term Project
CS 126 Term Project

- Imagine, Design, and Create your own android app.
- Three Requirements:
  - Have at least two activities (in the Android sense)
    - Today’s lecture
  - Have interactivity between App users across devices
    - Tuesday’s lecture
  - Use one Android platform feature not explicitly taught
    - E.g., hardware features (e.g., GPS, accelerometer, gyroscope, camera, microphone, speaker), authentication (e.g., Firebase), notifications (e.g., Firebase), activities with multiple fragments, custom Views using android.graphics.canvas,
Unix

- Compressed files
  - zip/unzip
  - gzip/gunzip
  - tar – create or read an archive

- Pipes
  - Feed the output of one command into another
Intents

- **Launch another App**
  - E.g., open a web page, Google Maps, send a text, phone call
  - Implicit Intent (specify task rather than App)

- **Launch another Activity in the same App.**
  - Main Activity -> Detail Activity
  - Explicit Intent (specify specific class to go to)
Passing Information with Intents

**Extras**
- Enables passing tagged, primitive data with an intent
  - The tag is a string
  - Only primitive types (int, String, etc); no objects

**Bundles**
- Android groups the extras together in a ‘Bundle’
- Serializes them (i.e., converts into a stream of bytes)
Sending non-primitives as Extras

- Parcelable interface
- Serialization


- Like converting to JSON and passing as String and converting back, but much more efficient.

- Addon to write the code for you. See:
Parcelable Example

```java
public class MyParcelable implements Parcelable {
    private int mData;
    public int describeContents() { return 0; }
    public void writeToParcel(Parcel out, int flags) {
        out.writeInt(mData);
    }
    public static final Parcelable.Creator<MyParcelable> CREATOR = new Parcelable.Creator<MyParcelable>() {
        public MyParcelablecreateFromParcel(Parcel in) {
            return new MyParcelable(in);
        }
        public MyParcelable[] newArray(int size) {
            return new MyParcelable[size];
        }
    };
    private MyParcelable(Parcel in) {
        mData = in.readInt();
    }
}
```