Week 16
Final Slides

CS 5254

End of semester activities

- **SPOT Survey** is open until Thursday (12/8) at 11:59pm ET
  - Please provide comments in addition to the scale answers
  - In return you will earn extra credit points for providing confirmation of submitting the survey
- **Final Exam** is open from Friday (12/9) at noon ET through Monday (12/12) at noon ET
  - Comprehensive, covering the entire semester
    - An emphasis on topics covered after the midterm (meaning Modules 7 through 11)
      - However, these generally build upon the foundations from Modules 1 through 6
  - The final exam is taken via Canvas, in exactly the same format as the midterm exam
  - The exam is split into 3 independent parts for scheduling convenience
    - Each part has 25 questions and a time limit of 40 minutes
    - As with the midterm, each part contains a randomized cross-section of topics

Semester topical review

- First Quarter
  - Android Development: Android, Android Studio, Kotlin, apps and layouts (P0)
  - Interactions & Activities: MVC architecture fundamentals, activity lifecycle (P1A)
  - ViewModels & Debugging: Device configuration changes, ViewModel lifecycle, Logcat (P1B)
  - Testing & Intents: Espresso instrumented tests, activity navigation via explicit intents (P1C)
  - Fragments, RecyclerViews & Interfaces: Fragment lifecycle, holders/adapters, data class (P2A)
  - Databases, Navigations & Dialogs: Coroutines, Room, navigation graphs, AlertDialog (P2B)
  - Midterm Exam
- Second Quarter
  - Menus & Implicit Intents: Menu providers, launching other apps via intents (P2C)
  - Photos & Swipes Gestures: Files and permissions, item touch helper (P2C)
  - Bottom Navigation & Network: Retrofit2 (REST), Moshi (JSON), and Coil (images) (P3)
  - Caching & WebViews: Moshi/Coil caching, HTML in Chrome, HTML in WebView (P3)
  - Maps & Map Markers: MapView via OSMDroid, map configurations, marker overlays (P3)
  - Final Exam (comprehensive)
- Important overarching themes running throughout the entire semester:
  - MVC separation of concerns
  - Object-oriented best practices (especially encapsulation)
  - Functional programming techniques

Thank You!