



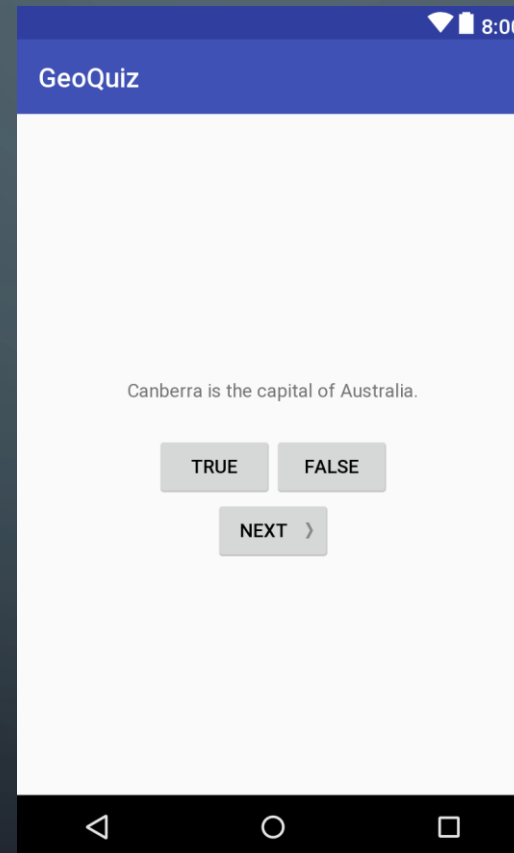
ANDROID AND MVC

Cs 347 Mobile Application Development

Instructor: Philip F. Garofalo, m.s.

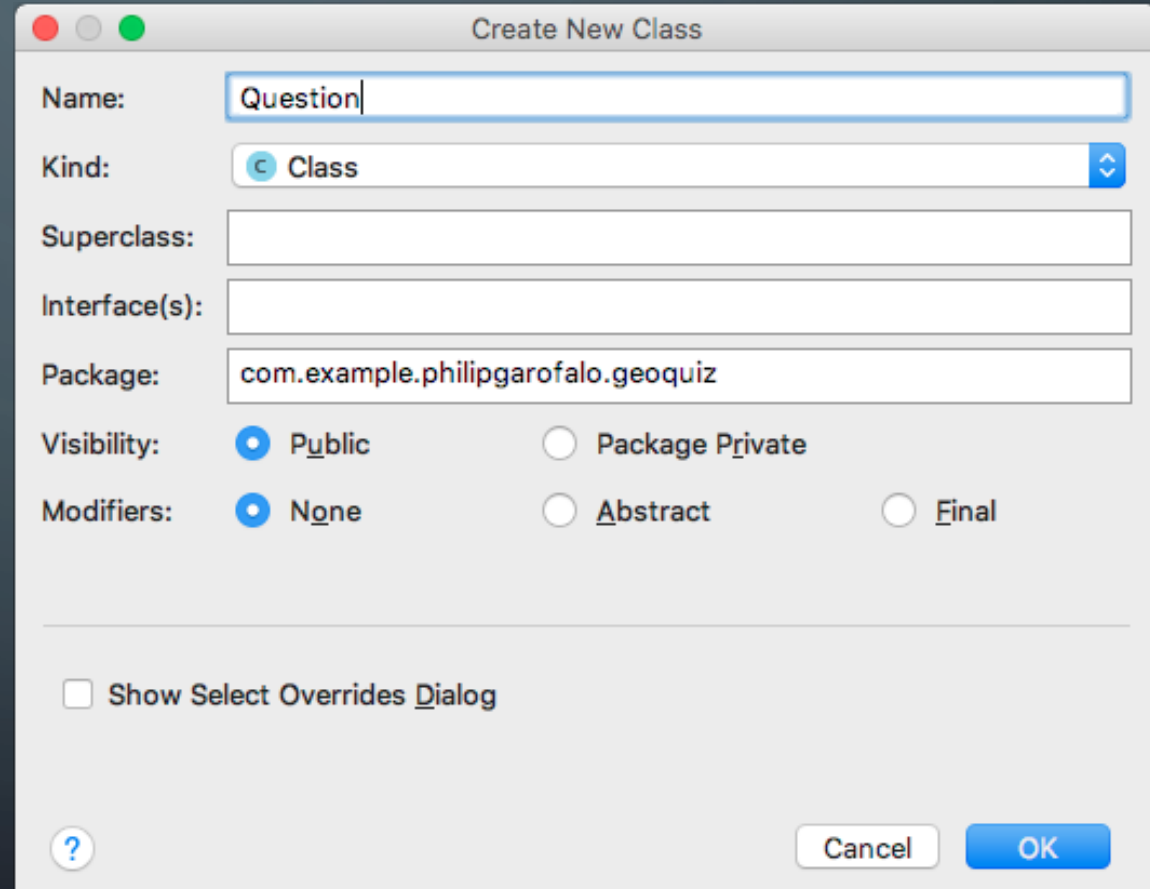
NEIU

GEOQUIZ 2: MULTIPLE QUESTIONS



CREATING A NEW CLASS

Right click on the page name in the project tool window and select New>Java Class. Name the class “Question” and click OK.



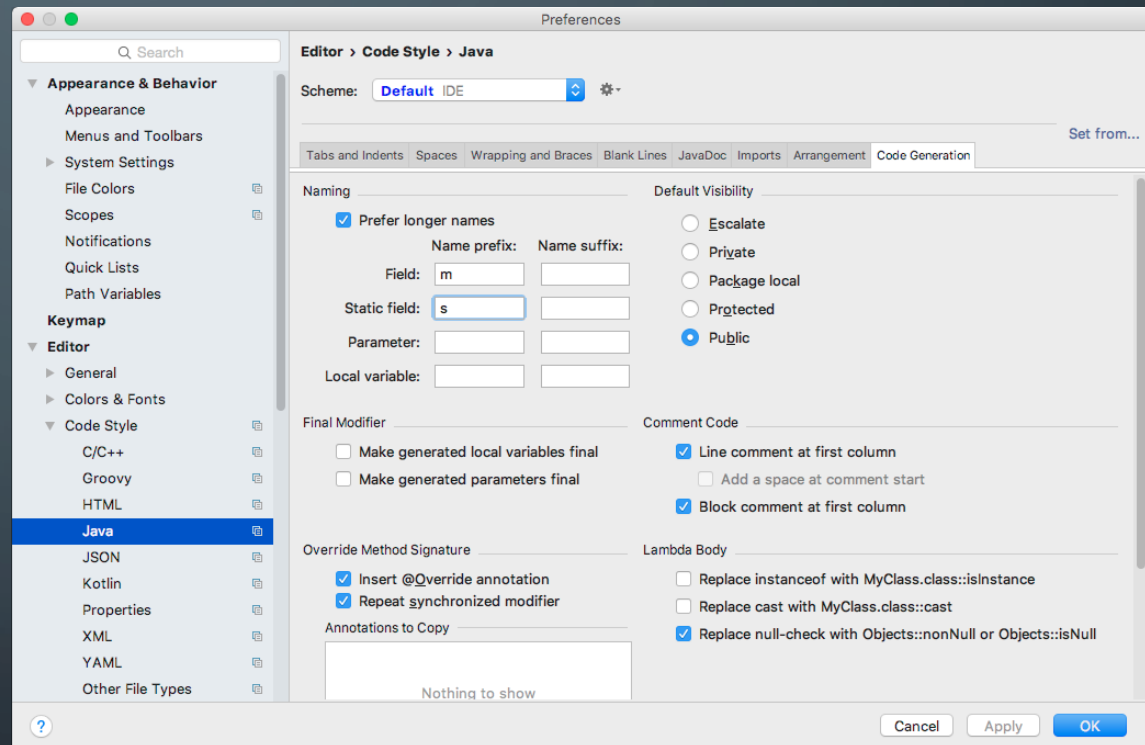
QUESTION CLASS

Add two instance variables and a constructor to the Question class.

```
public class Question {  
  
    private int mTextResId;  
    private boolean mAnswerTrue;  
  
    public Question(int textResId, boolean answerTrue) {  
        mTextResId = textResId;  
        mAnswerTrue = answerTrue;  
    }  
}
```

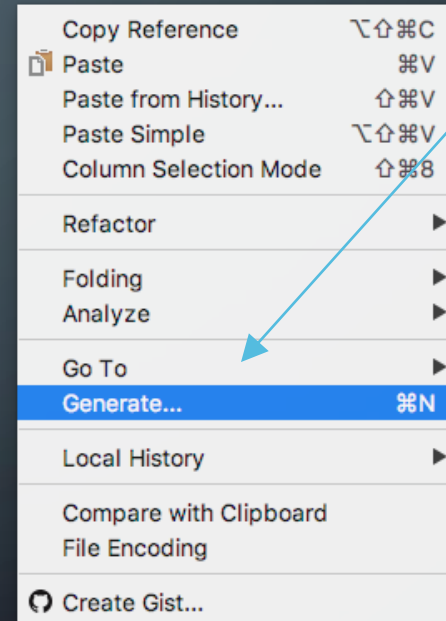
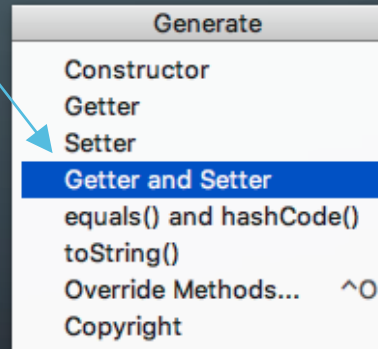
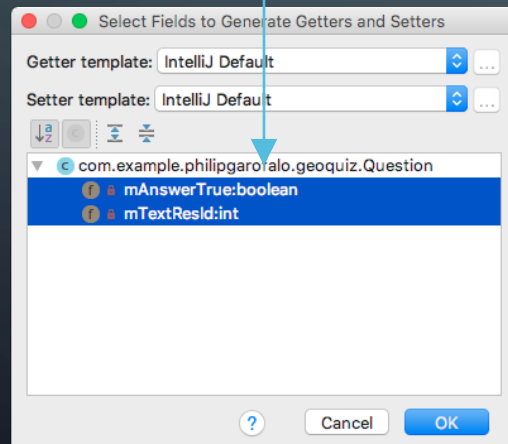
GENERATING GETTERS AND SETTERS

- Open Settings
 - Mac: Android Studio>Preferences
 - On Windows: File>Settings
- Configure Android Studio to recognize the m prefix for member instance variables.
- Add the s prefix for static variables.
- Click OK.



GENERATING GETTERS AND SETTERS (PART 2)

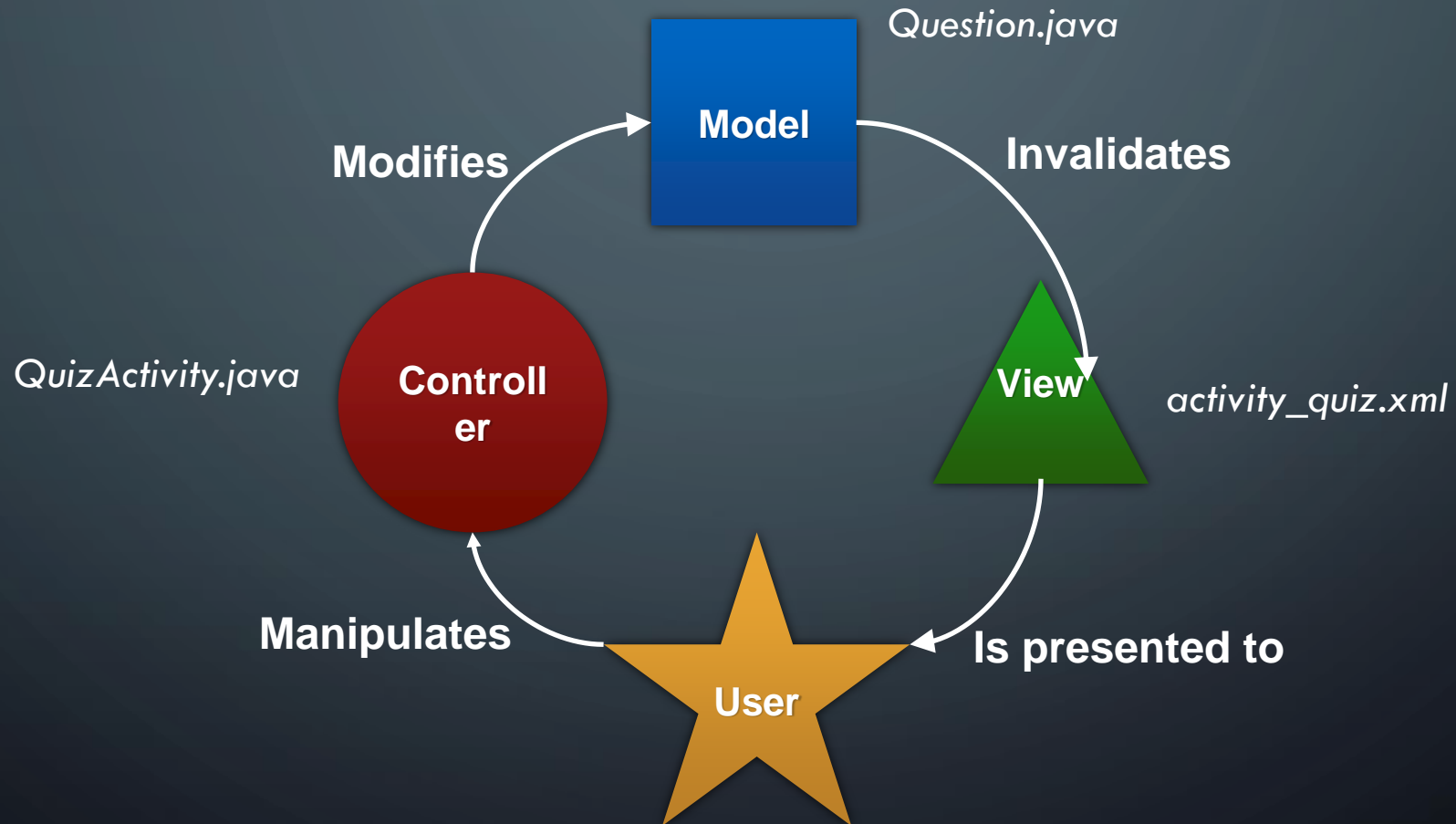
- Back in Question.java, right click after the constructor and select **Generate...**
- Then select **Getter and Setter**.
- Select the two data members.
- Click OK.



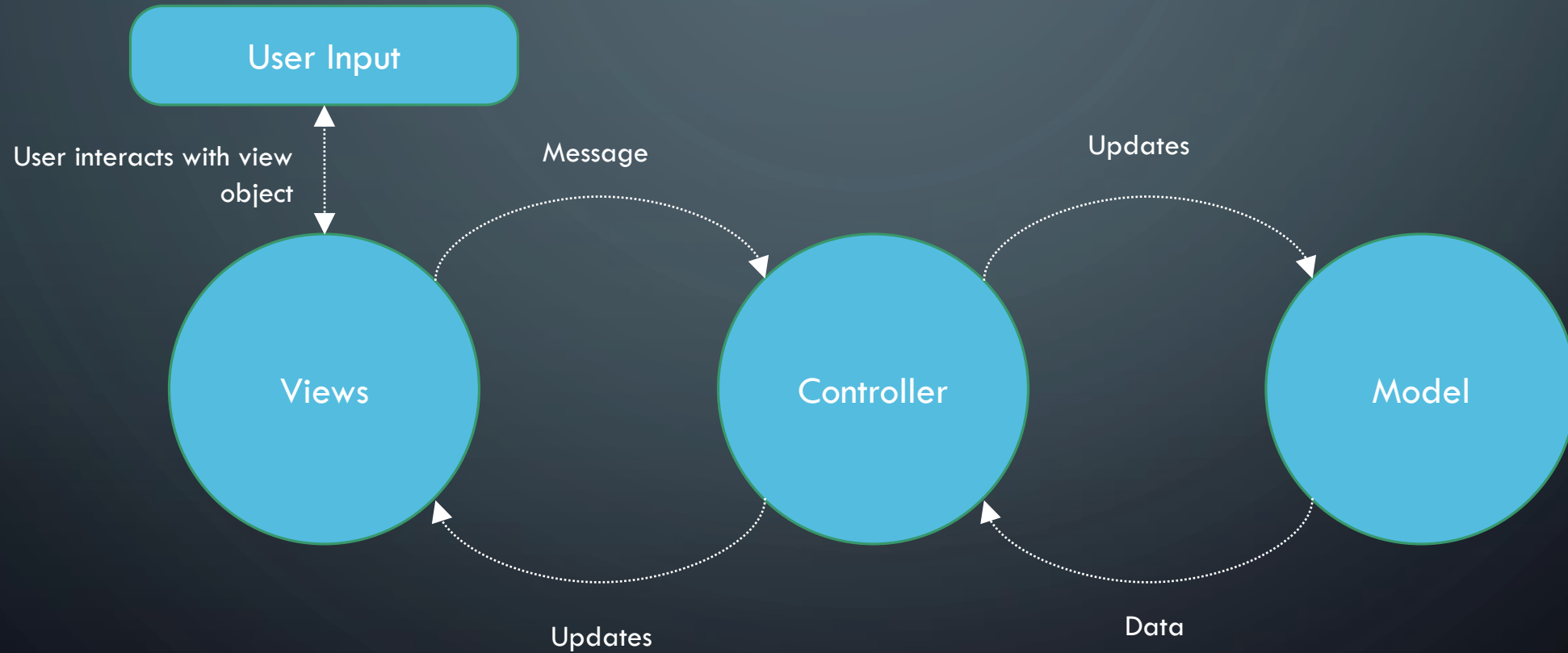
GENERATING GETTERS AND SETTERS (PART 3)

```
public class Question {  
  
    private int mTextResId;  
    private boolean mAnswerTrue;  
  
    ...  
  
    public int getTextResId() {  
        return mTextResId;  
    }  
  
    public void setTextResId(int textResId) {  
        mTextResId = textResId;  
    }  
  
    public boolean isAnswerTrue() {  
        return mAnswerTrue;  
    }  
  
    public void setAnswerTrue(boolean answerTrue) {  
        mAnswerTrue = answerTrue;  
    }  
}
```

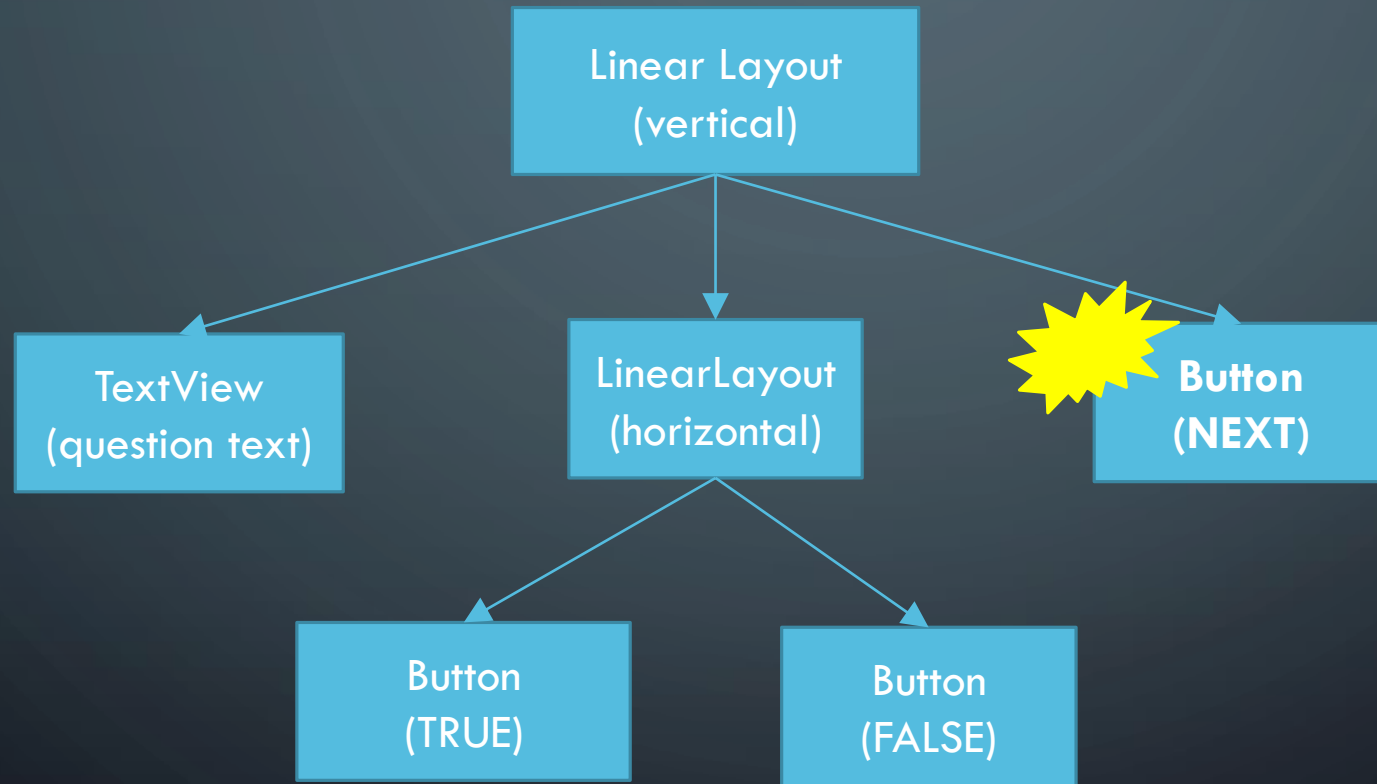
MODEL-VIEW-CONTROLLER



MODEL-VIEW-CONTROLLER IN ANDROID



ADD THE NEXT BUTTON



UPDATING THE VIEW LAYER

- Make the the question text changeable.
- We'll remove the TextView's hardcoded `android:text` attribute.
- We'll give it a resource ID with the `android:id` attribute.
- Add the Next button the root vertical `LinearLayout`.

UPDATING THE VIEW LAYER

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:gravity="center" >
```

```
<TextView
    android:id="@+id/question_text_view"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:padding="24dp" />
```

```
<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal">
```

```
<Button
    android:id="@+id/true_button"
```

```
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/true_button" />
```

```
<Button
    android:id="@+id/false_button"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:onClick="onFalseClick"
    android:text="@string/false_button" />
```

```
</LinearLayout>
```

```
<Button
    android:id="@+id/next_button"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/next_button" />
```

```
</LinearLayout>
```

ADD QUIZ QUESTIONS

- res/values/strings.xml

```
<resources>
  <string name="app_name">GeoQuiz</string>
  <string name="true_button">True</string>
  <string name="false_button">False</string>
  <string name="question_australia">Canberra is the capital of Australia.</string>
  <string name="correct_toast">Correct!</string>
  <string name="incorrect_toast">Incorrect!</string>
  <string name="next_button">Next</string>
  <string name="question_oceans">The Pacific Ocean is larger than the Atlantic Ocean</string>
  <string name="question_mideast">The Suez Canal connects the Red Sea and the Indian Ocean</string>
  <string name="question_africa">The source of the Nile River is in Egypt.</string>
  <string name="question_americas">The Amazon River is the longest river in the Americas.</string>
  <string name="question_asia">Lake Baikal is the world\'s oldest and deepest freshwater lake.</string>
</resources>
```

UPDATING THE CONTROLLER LAYER

- Add three instance variables and the question array to the QuizActivity class

```
public class QuizActivity extends AppCompatActivity {  
  
    private Button mTrueButton;  
    private Button mFalseButton;  
    private Button mNextButton;  
    private TextView mQuestionTextView;  
  
    private Question[] mQuestionBank = new Question[] {  
        new Question(R.string.question_australia, true),  
        new Question(R.string.question_oceans, true),  
        new Question(R.string.question_mideast, false),  
        new Question(R.string.question_africa, false),  
        new Question(R.string.question_americas, true),  
        new Question(R.string.question_asia, true)  
    };  
  
    private int mCurrentIndex = 0;  
}
```

UPDATING THE CONTROLLER LAYER (PART 2)

- Wire up the TextView. Add the following lines to `QuizActivity.onCreate()`.

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_quiz);

    mQuestionTextView = (TextView) findViewById(R.id.question_text_view);
    int question = mQuestionBank[mCurrentIndex].getTextResId();
    mQuestionTextView.setText(question);

    mTrueButton = (Button) findViewById(R.id.true_button);
```

UPDATING THE CONTROLLER LAYER (PART 3)

- Wire up the Next button. Add the following lines at the end of `QuizActivity.onCreate()`.

```
mFalseButton = (Button) findViewById(R.id.false_button);

mNextButton = (Button) findViewById(R.id.next_button);
mNextButton.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        mCurrentIndex = (mCurrentIndex + 1) % mQuestionBank.length;
        int question = mQuestionBank[mCurrentIndex].getTextResId();
        mQuestionTextView.setText(question);
    }
});
```


UPDATING THE CONTROLLER LAYER (PART 4)

- Refactor redundant lines of code in onCreate() as follows and then *run* the app.

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    ...
    mQuestionTextView = (TextView) findViewById(R.id.question_text_view);
    // two lines removed from here.
    ...
    mNextButton.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            mCurrentIndex = (mCurrentIndex + 1) % mQuestionBank.length;
            // two lines removed from here.
            UpdateQuestion();
        }
    });

    UpdateQuestion();
}

private void UpdateQuestion() {
    int question = mQuestionBank[mCurrentIndex].getTextResId();
    mQuestionTextView.setText(question);
}
...
```

UPDATING THE CONTROLLER LAYER (PART 5)

- Add a method to check answers in QuizActivity.

```
private void checkAnswer(boolean userChoice) {  
    boolean correctAnswer = mQuestionBank[mCurrentIndex].isAnswerTrue();  
  
    int messageResId = 0;  
  
    if (userChoice == correctAnswer) {  
        messageResId = R.string.correct_toast;  
    } else {  
        messageResId = R.string.incorrect_toast;  
    }  
  
    Toast.makeText(this, messageResId, Toast.LENGTH_SHORT).show();  
}
```

UPDATING THE CONTROLLER LAYER (PART 6)

- Change the button click listeners to call `checkAnswer()`. Then run the app.

```
...
mTrueButton.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        checkAnswer(true);
    }
});
...
public void onFalseClick(View v) {
    checkAnswer(false);
}
```

RUNNING ON A DEVICE

- Enable your Android device for debugging by opening the **Settings**, tapping **About**, *and then tapping on the build number seven times*.
- Connect your device to your laptop with its **micro-USB cable**.
- If your Windows laptop doesn't recognize the device, you will need to download and install the **Android Debug Bridge (adb)**.
 - <https://software.intel.com/en-us/xdk/docs/installing-android-debug-bridge-adb-usb-driver-on-windows>
 - <https://developer.android.com/studio/run/oem-usb.html>

ADDING AN ICON TO A BUTTON

- Two ways: copy and paste or Android Asset Studio (built-in to Android Studio).
- Text uses copy and paste.
 - Download bignerdranch.com/solutions/AndroidProgramming3e.zip (it's in D2L too)
 - Copy the four “drawable” folders in 02_MVC/GeoQuiz/app/src/main/res.
 - Paste them into app/res directory in the Project tool window.

ANDROID PIXEL DENSITIES

Suffix	Meaning
ldpi	Resources for low-density (<i>ldpi</i>) screens (~120dpi).
mdpi	Resources for medium-density (<i>mdpi</i>) screens (~160dpi). (This is the baseline density.)
hdpi	Resources for high-density (<i>hdpi</i>) screens (~240dpi).
xhdpi	Resources for extra-high-density (<i>xhdpi</i>) screens (~320dpi).
xxhdpi	Resources for extra-extra-high-density (<i>xxhdpi</i>) screens (~480dpi).
xxxhdpi	Resources for extra-extra-extra-high-density (<i>xxxhdpi</i>) uses (~640dpi).
nodpi	Resources for all densities. These are density-independent resources. The system does not scale resources tagged with this qualifier, regardless of the current screen's density.

ADDING AN ICON TO A BUTTON (PART 2)

...

```
<Button
    android:id="@+id/next_button"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:drawablePadding="4dp"
    android:drawableRight="@drawable/arrow_right"
    android:text="@string/next_button" />
</LinearLayout>
```

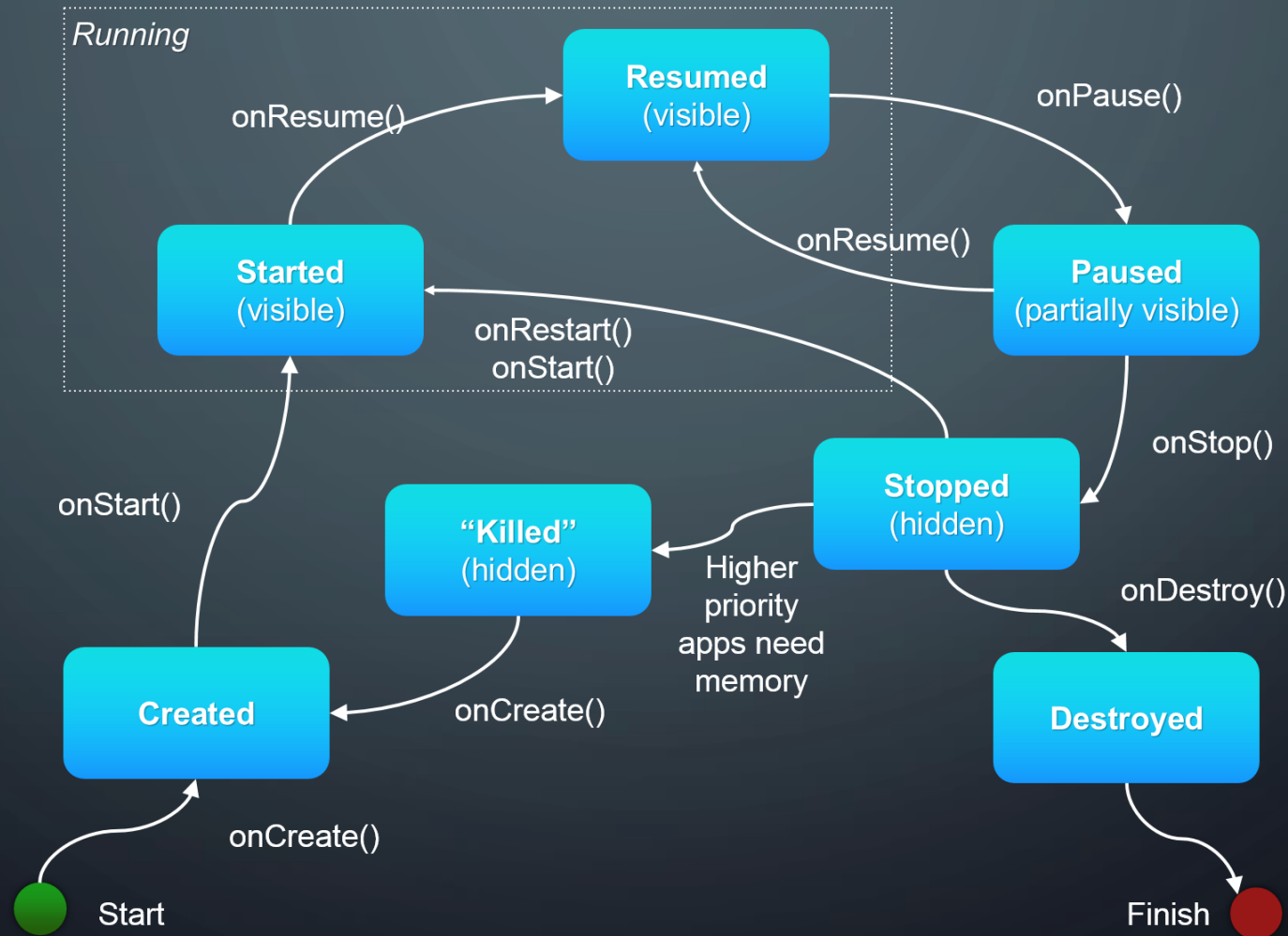
RUN GEOQUIZ 2!

- Take the quiz and go through the answers.
- Homework: implement the Challenge “Add a Previous Button”

THE ACTIVITY LIFECYCLE

- When you running GeoQuiz and you rotate your device, the question reverts to the first question.
- How did that happen?
- How can we fix it?

ACTIVITY STATES



ACTIVITY STATES (PART 2)

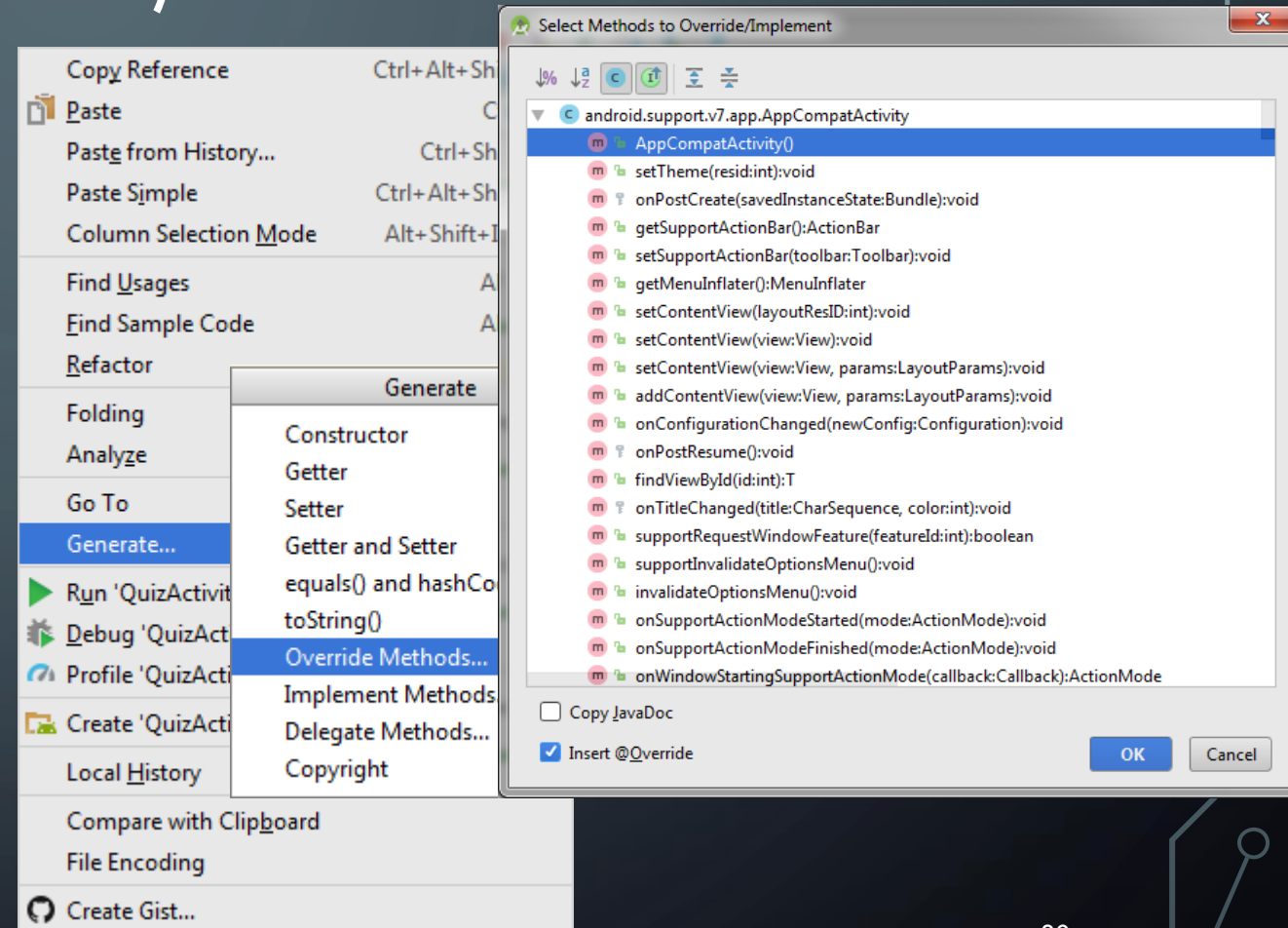
State	In memory?	Visible to user?	In foreground?
Nonexistent	No	No	No
Stopped	Yes	No	No
Paused	Yes	Partially	No
Started/Resumed	Yes	Yes	Yes

ADDING LOGGING

```
public class QuizActivity extends AppCompatActivity {  
    private static final String TAG = "QUIZACTIVITY";  
    ...  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        Log.d(TAG, "onCreate");  
        setContentView(R.layout.activity_quiz);  
    }  
}
```

ADDING LOGGING (PART 2)

- Generate overrides for state transition methods:
- Right click in QuizActivity, outside of any method and select **Generate...**
- Then select **Override Methods...**
- Then select onStart(), onStop(), onPause(), onResume(), and onDestroy().
- Click **OK**.



ADDING LOGGING (PART 3)

```
@Override
protected void onStart() {
    super.onStart();
    Log.d(TAG, "onStart: ");
}

@Override
protected void onStop() {
    super.onStop();
    Log.d(TAG, "onStop: ");
}

@Override
protected void onDestroy() {
    super.onDestroy();
```

```
    Log.d(TAG, "onDestroy: ");
}
```

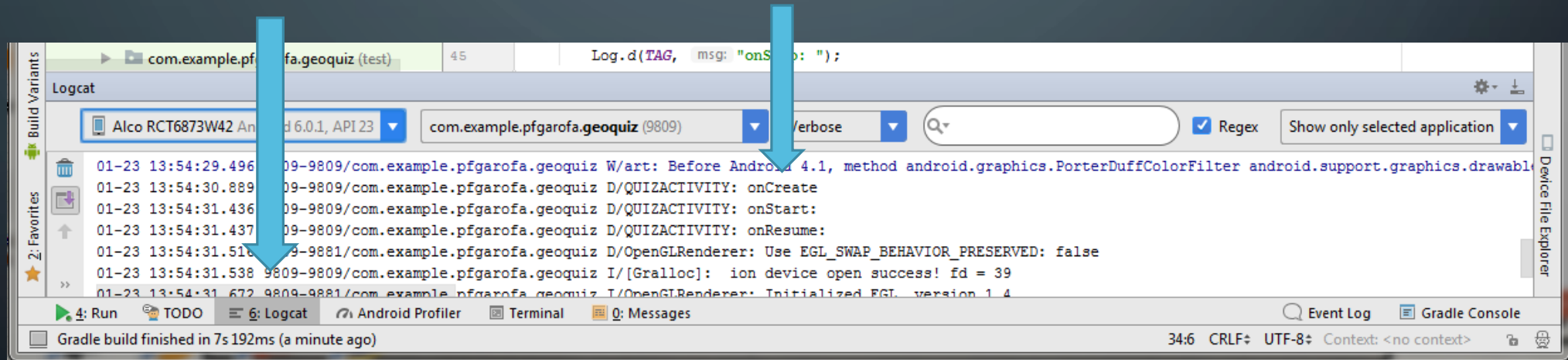
```
@Override
protected void onPause() {
    super.onPause();
    Log.d(TAG, "onPause: ");
}
```

```
@Override
protected void onResume() {
    super.onResume();
    Log.d(TAG, "onResume: ");
}
```

```
}
```

LOGCAT

- Now run GeoQuiz.
- Open the LogCat console and see the log messages:



LOGGING LEVELS

- `Log.v(TAG, message)` - verbose
- `Log.d(TAG, message)` - debug
- `Log.i(TAG, message)` - information
- `Log.w(TAG, message)` - warning
- `Log.e(TAG, message)` - error
- `Log.wtf(TAG, message)` - What a Terrible Failure!

DEVICE ROTATION

- When an Android device is rotated the current activity is **destroyed**!
- And a new one is **created**!
- That's why the first question comes back.

CREATE A LANDSCAPE LAYOUT

- When you rotate your device, the **device configuration** changes.
- When the device configuration changes, Android looks for **alternative resources** to use that more closely match the new configuration.
- So let's create a landscape layout.
- Open the layout select the “rotated device” icon and select **Create Landscape Variation...**

CREATE A LANDSCAPE LAYOUT (PART 2)

PRESERVING DATA ACROSS ROTATIONS

DEBUGGING ANDROID APPS

EXCEPTIONS

STACK TRACES



BREAKPOINTS

EXCEPTION BREAKPOINTS



LINT

R CLASS DEBUGGING