

- An Activity goes through a life cycle, and methods are called automatically as an activity is started, paused, stopped, or closed.
- There are many ways to either pass data or share data between two activities, which include: using the `putExtra` methods of the `Intent` class, using a singleton class for the Model, or using a global variable representing the Model.
- One way for activities to share data is to declare a `public static` instance of a class of the Model in one Activity class. In this way, it is global and can be accessed by any other Activity class.
- A transition is an animation special effect going from one screen to another.
- The android framework provides classes for fading, scaling, translating, and rotating animations. Animations can be coded in XML files and placed in the `anim` directory, which should be placed in the `res` directory.
- The `SharedPreferences` interface provides the functionality to write and read preferences to the file system.
- The `getDefaultSharedPreferences` static method of the `PreferenceManager` class returns a `SharedPreferences` reference.
- If an app writes to the file system, we need to include a `uses-permission` element in the `AndroidManifest.xml` file.



Exercises, Problems, and Projects

Multiple-Choice Exercises

1. The `TableLayout` class can be used to organize various GUI components
 - ✓ • As a table of rows and columns
 - As a table of multiple rows with only one column each
 - As a table of only one row and multiple columns
 - As a table of only one row and one column
2. The direct superclass of `LinearLayout` and `RelativeLayout` is
 - View
 - ✓ • ViewGroup
 - Layout
 - Object

3. `TableLayout` and `TableRow` are direct subclasses of
 - ✓ `LinearLayout`
 - `ViewGroup`
 - `RelativeLayout`
 - `View`
4. The `RelativeLayout` class is a good choice to organize various GUI components
 - To give the components absolute x- and y-coordinates
 - ✓ So that we position components relative to other components
 - As a grid of multiple rows and columns
 - It is never a good choice
5. In what package is the `Intent` class?
 - `java.intent`
 - `android.widget`
 - `android.activity`
 - ✓ `android.content`
6. After you have created an `Intent` for a new activity, what method of the `Activity` class do you call with that `Intent` parameter in order to start a new activity?
 - ✓ `startActivity`
 - `newActivity`
 - `startIntent`
 - `newIntent`
7. What method of the `Activity` class is automatically called when an activity is about to restart?
 - `onCreate`
 - `onDestroy`
 - ✓ `onRestart`
 - `onGo`
8. What methods of the `Activity` class (and in what order) are automatically called when an activity is first created?
 - `onCreate`
 - ✓ `onCreate, onStart, and onResume` (in that order)
 - `onCreate and onResume`
 - `onStart, onCreate, and onResume` (in that order)
9. What method of the `Activity` class is automatically called when an activity becomes invisible to the user?
 - `onResume`
 - ✓ `onStop`
 - `onPause`
 - `onInvisible`

10. Two activities can share the same data
- No, it is not possible
 - Yes, but it is only possible by writing to and reading from the same file
 - Yes, but it is only possible by writing to and reading from a SQLite database
 - ✓ Yes, for example by each accessing a public static instance variable from another class
11. In what package do we find the Animation class?
- android.animation
 - android.view
 - ✓ android.view.animation
 - android.animation.view
12. What is not a subclass of the Animation class?
- ScaleAnimation
 - RotateAnimation
 - AlphaAnimation
 - ✓ MoveAnimation
13. What class do we use to play several animations together?
- AnimationSequence
 - SequenceAnimation
 - SeveralAnimation
 - ✓ AnimationSet
14. What static method of the class PreferenceManager do we use to get SharedPreferences?
- getPreferences
 - sharedPreferences
 - getDefaultPreferences
 - ✓ getDefaultSharedPreferences

Fill in the Code

15. Inside a TableRow element, this code adds a row that contains an EditText and a TextView whose ids are game and player.

```
<TableRow
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" >
    <!--Your code goes here -->
```

```
</TableRow>
```