

CS E-76: Building Mobile Applications

Intents, Menus, and
SharedPreferences

TF: Sophie Chang

Using intents to start a second Activity

- Remember to add new Activity(s) to AndroidManifest.xml !
- Intent intent = new Intent(Context packageContext, Class<?> class);
 - packageContext = the activity we're coming from
 - Class: the Activity in our app we're going to
- Intent intent = new Intent(this, AnotherActivity.class);
- startActivity(i);

Requesting and Starting Another App's Activity

- `Intent intent = new Intent(<action>, <data>);`
 - `ACTION_VIEW`, `ACTION_EDIT`, `ACTION_DIAL`
 - `scheme://host/data/segments`
- `Intent intent = new Intent(Intent.ACTION_DIAL, Uri.parse("tel:1234567890"));`

Activity Results

- `startActivityForResult(intent)` allows us to “return” data to the calling Activity
- `onActivityResult(int requestCode, int resultCode, Intent data);`
 - `requestCode`: unique integer to determine what Activity returned
 - `resultCode`: `RESULT_OK`, `RESULT_CANCELED`
 - `data`: Intent to receive additional data

Activity Results (cont'd)

- setResult(int resultCode, Intent data);
 - resultCode: RESULT_OK, RESULT_CANCELED
 - data: Intent to pass additional data
- finish() pops from stack and sends control back to previous Activity

Bundles

- Bundle objects store key and value pairs via the `put<type>` methods and `get<type>` methods
 - Basic data types: Boolean, Byte, Char, Double, Float, Int, Long, etc.
 - Non-primitives: String, Serializable, Arrays
- Can get `size()` and `keySet()`

Bundles (cont'd)

- `intent.putExtras(Bundle bundle);`
 - Associate a bundle with an intent
- `getIntent().getExtras();`
 - Get bundle associated with an intent
- `intent.putExtra(String key, value);`
 - `intent.getExtra(String key);`

Menus

- Described in XML, stored in res/menu
- `<menu>` has many `<item>` children
- Accessed like other resources
(`R.menu.<filename>`)

Menu (cont'd)

- `onCreateOptionsMenu(Menu menu)`
 - User presses menu button on device
- `onCreateContextMenu(ContextMenu menu, View v, ContextMenuInfo menuInfo)`
 - User holds down a view (right-clicking/long press)
 - Need to `registerForContextMenu(View v)`

Menus

- `onOptionsItemSelected()` (MenuItem item)
 - User selects an option from the menu coming from the menu button
- `onContextItemSelected()` (MenuItem item)
 - User selected an option from a context menu
- `Item.getItemId()`
 - Gets ID of selected item

Submenus

- `<item>` elements can also contain other `<menu>` items!
- When `<item>` is selected, a submenu will open as a context menu

SharedPreferences

- Lightweight key-value pair storage
- General framework that allows you to save and retrieve persistent key-value pairs of primitive data types.
- Use it to save any primitive data: booleans, floats, ints, longs, and strings.
- Persist amongst user sessions

Get your SharedPreferences

- `getSharedPreferences()` – Use this if you need multiple preferences files identified by name where you specify with the first parameter
- `getPreferences()` – Use this if you need only one preferences file for your Activity. Since this will be the only preferences file for your Activity, you don't supply a name

Write to your SharedPreferences

- Get a SharedPreferences.Editor
 - Call edit()
 - Add values with methods (put<type>())
 - Commit new values with commit()

Read from your SharedPreferences

- Use SharedPreferences methods (`get<type>()`) to grab the values from your preferences
- EX: `getBoolean(key, defaultvalue);`