Native Android Actions

Native Android applications also use Intents to launch Activities and sub-Activities. Here are some of the native actions available as static string constant in the Intent class. When creating implicit Intents you can use these actions.

- **ACTION_CALL** Brings up a phone dialer and **immediately** initiates a call using the number supplied in the Intent URI
- **ACTION_DIAL** Brings up a dialer application with the number to dial pre-populated from the Intent URI for example tel:555-123.
- **ACTION_PICK** Launches a sub-Activity that lets you pick an item from the Content Provider specified by the Intent URI. When closed it should return a URI to the item that was picked. The Activity launched depends on the data being picked: for example, passing content://contacts/people will invoke the native contacts list.
- **ACTION_VIEW** The most common generic action. View asks that the data supplied in the Intent’s URI be viewed in the most reasonable manner.
- **ACTION_SENDTO** Launches an Activity to send a message to the contact specified by the Intent URI.
- **And many others –some of them in our book,, page :143,144-**

Natively **http** addresses will open in the browser, **tel**: addresses will open the dialer to call the number, **geo**: addresses will be displayed in the Google Maps application, and contact content will be displayed in the contact manager.

### Lab Work 1

1. Create an activity with an EditText and 3 buttons.
2. Buttons jobs as follows:
   - First button calls the number in the EditText. (**need permission!!**)
   - Second Button Views a website in the Edit Text.
   - Third Button sends the number a message.
Using Intent Filters to Service Implicit Intents

Using Intent Filters, application components announce that they can respond to action requests from any application installed on the device.

To register an application component as a potential Intent handler, add an intent-filter tag to the component’s manifest node using the following tags (and associated attributes) within the Intent Filter node:

1. Action Uses the android:name attribute to specify the name of the action being serviced.
2. Category Uses the android:name attribute to specify under which circumstances the action should be serviced.
   - DEFAULT Set this to make a component the default action for the data type specified
   - LAUNCHER Using this category makes an Activity appear in the application Launcher.
3. data The data tag lets you specify which data types your component can act on;
   You can use any combination of the following attributes to specify the data your component supports:
   - android:scheme Requires a particular scheme (e.g., content or http).

Filter Examples

```xml
<activity android:name=".IntentFilters">
    <intent-filter>
        <action android:name="android.intent.action.MAIN"/>
        <category android:name="android.intent.category.LAUNCHER"/>
    </intent-filter>
</activity>

<activity android:name="MyBrowser">
    <intent-filter>
        <action android:name="android.intent.action.VIEW"/>
        <category android:name="android.intent.category.DEFAULT"/>
        <data android:scheme="http"/>
    </intent-filter>
</activity>
```
<table>
<thead>
<tr>
<th>Lab Work 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Construct an activity that can handle http websites.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Homework</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Later</td>
</tr>
</tbody>
</table>