

Разработка мобильных приложений

Android


2

План

- Чему я сегодня научусь?
 - Поддерживать разные размеры экранов, платформы устройств, языки
 - Подключаться к интернету
 - Создавать меню
- Про что я сегодня узнаю?
 - ActionBar
 - AsyncTask
 - Уровни log-сообщений

Log

Уровни log-сообщений

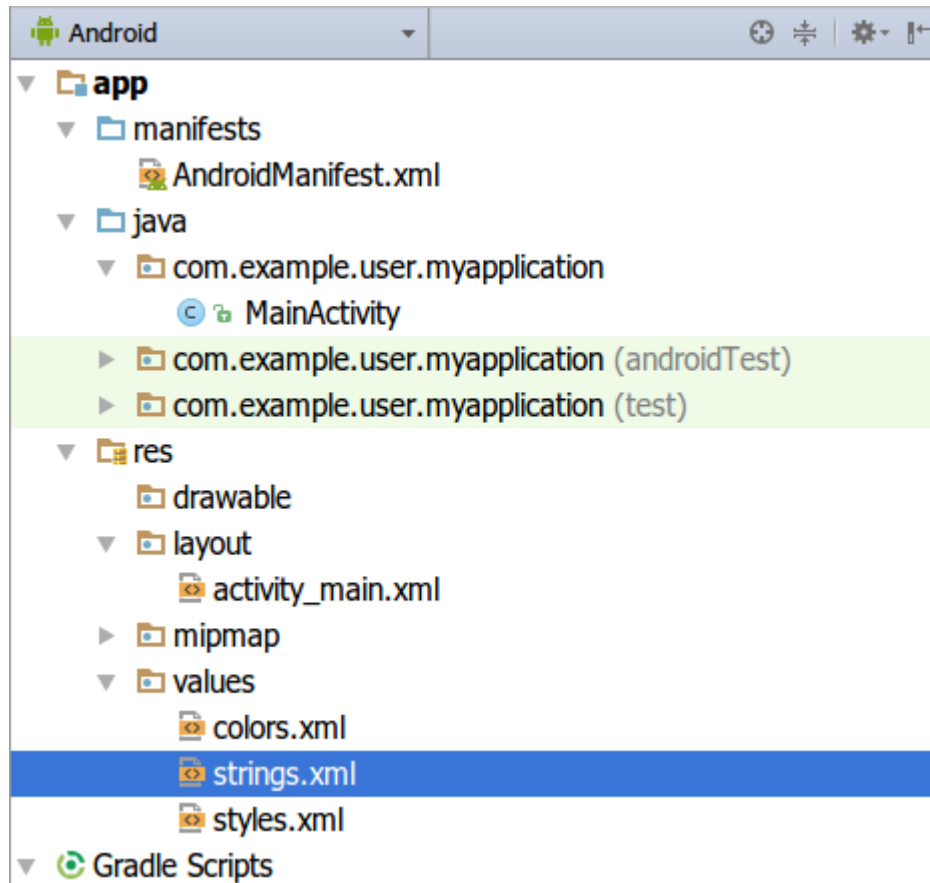
- verbose
 - debug
 - info
 - warn
 - error
 - [WTF] — What a Terrible Failure (assert)
- 
- Для отладки

Пример:

```
private static final String TAG = "MyActivity";  
...  
Log.i(TAG, "activity created");
```

Типы ресурсов

<https://developer.android.com/guide/topics/resources/providing-resources.html>



res/values/strings.xml

activity_main.xml × MainActivity.java × app × strings.xml × Translations Editor ×

Edit translations for all locales in the translations editor. [Open editor](#) [Hide notification](#)

```
13  See the License for the specific language governing permissions and
14  limitations under the License.
15  -->
16  <resources>
17      <string name="app_name">Github Query</string>
18      <string name="button_enter_pass">Enter Password</string>
19  </resources>
20
```

in.xml × MainActivity.java × app × strings.xml × Translations Editor × 1

[Order a translation...](#)

Key	Untransl...	Default Value
app_name	<input type="checkbox"/>	Github Query
button_enter_pass	<input type="checkbox"/>	Enter Password

Key:

Default Value:

Resource Folder:

[OK](#) [Cancel](#)

Key:

Default Value:

Translation:

Event Log Gradle Console

1:15 n/a n/a Context: <no context>

res/values/strings.xml

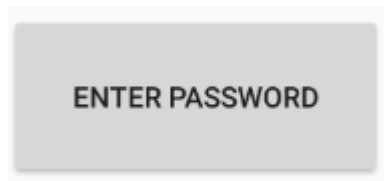
Обращение к строкам, заданным в strings.xml:

- Из .xml файла:

```
<Button  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="@string/button_enter_pass"  
    android:padding="32dp"/>
```

- Из .java:

```
String buttonText = getString(R.string.button_enter_pass);  
  
Button enterPassButton = (Button) findViewById(R.id.enter_pass_button);  
enterPassButton.setText(buttonText);
```

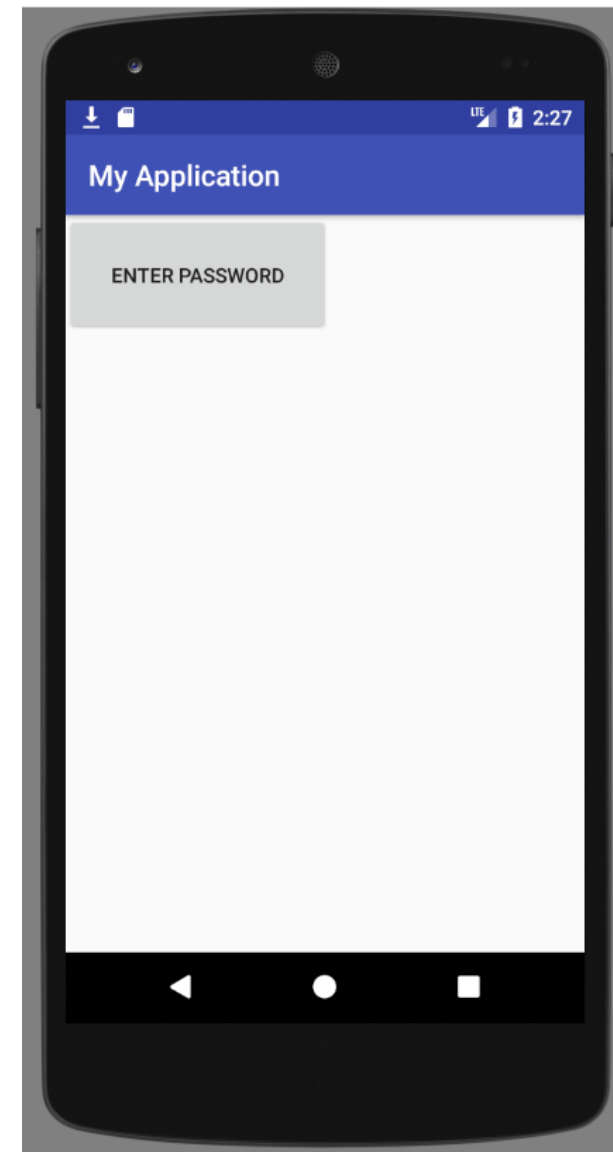


Обращение к ресурсам

- **setContentView**(R.layout.layout_file_name)
- **findViewById**(R.id.some_id)
- **getString**(R.string.string_name)

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3   android:layout_width="match_parent"
4   android:layout_height="match_parent"
5   android:orientation="vertical">
6   <Button
7     android:id="@+id/enter_pass_button"
8     android:layout_width="wrap_content"
9     android:layout_height="wrap_content"
10    android:padding="32dp"/>
11 </LinearLayout>
```

```
1 package com.example.user.myapplication;
2
3 import ...
4
5
6
7
8
9 public class MainActivity extends AppCompatActivity {
10
11     @Override
12     protected void onCreate(Bundle savedInstanceState) {
13         super.onCreate(savedInstanceState);
14         setContentView(R.layout.activity_main);
15
16
17         String buttonText = getString(R.string.button_enter_pass);
18
19         Button enterPassButton =
20             (Button)findViewById(R.id.enter_pass_button);
21         enterPassButton.setText(buttonText);
22     }
23 }
```



Поддержка разных платформ

- minSDK, targetSDK
- Проверка системной версии во время выполнения

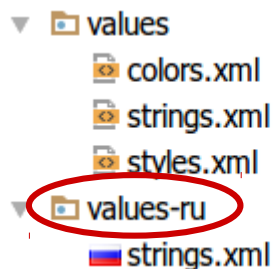
```
private void setUpActionBar() {  
    // Make sure we're running on Honeycomb or higher to use ActionBar APIs  
    if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.HONEYCOMB) {  
        ActionBar actionBar = getActionBar();  
        actionBar.setDisplayHomeAsUpEnabled(true);  
    }  
}
```

- Использование системных тем и стилей

```
<activity android:theme="@android:style/Theme.Translucent">
```


Поддержка языков

- Создание директории для другого языка в формате dirname-country
- Создание одноименных файлов ресурсов с ресурсами для другого языка



res/values/strings.xml

```
1 <resources>
2   <string name="app_name" translatable="false">My Application</string>
3   <string name="button_enter_pass">Enter Password</string>
4 </resources>
```

res/values-ru/strings.xml

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <resources>
3   <string name="button_enter_pass">Введите пароль</string>
4 </resources>
```

```
Resources resources = getResources();
Configuration configuration = resources.getConfiguration();
configuration.setLocale(new Locale("ru"));
resources.updateConfiguration(configuration, null);
/*if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.JELLY_BEAN_MR1) {
    createConfigurationContext(configuration);
} else {
    resources.updateConfiguration(configuration, null);
}*/
String buttonText = getString(R.string.button_enter_pass);
```

Поддержка разных экранов

```
MyProject/  
  res/  
    layout/                # default (portrait)  
      main.xml  
    layout-land/           # landscape  
      main.xml  
    layout-large/          # large (portrait)  
      main.xml  
    layout-large-land/     # large landscape  
      main.xml
```

Добавление директорий с ресурсами

res → New → Resource Directory

New Resource Directory

Directory name: values-ar

Resource type: values

Source set: main

Available qualifiers:

- Country Code
- Network Code
- Layout Direction
- Smallest Screen Width
- Screen Width
- Screen Height
- Size
- Ratio
- Orientation
- UI Mode
- Night Mode
- Density

Chosen qualifiers:

- ar

Language:

- aa: Afar
- ab: Abkhazian
- ae: Avestan
- af: Afrikaans
- ak: Akan
- am: Amharic
- an: Aragonese
- ar: Arabic
- as: Assamese
- av: Avaric
- ay: Aymara

Specific Region Only:

- Any Region
- AE: United Arab Emirates
- BH: Bahrain
- DJ: Djibouti
- DZ: Algeria
- EG: Egypt
- EH: Western Sahara
- ER: Eritrea
- IL: Israel
- IQ: Iraq
- JO: Jordan

Tip: Type in list to filter

☐ Show All Regions

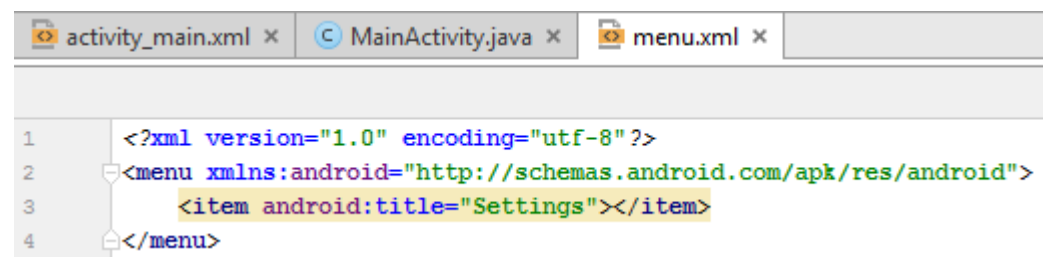
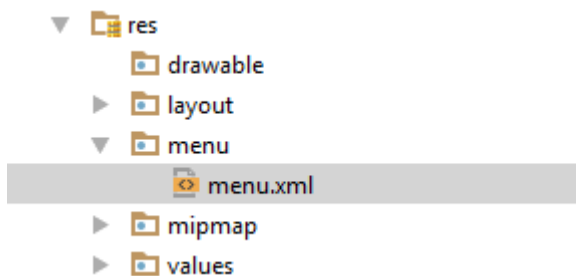
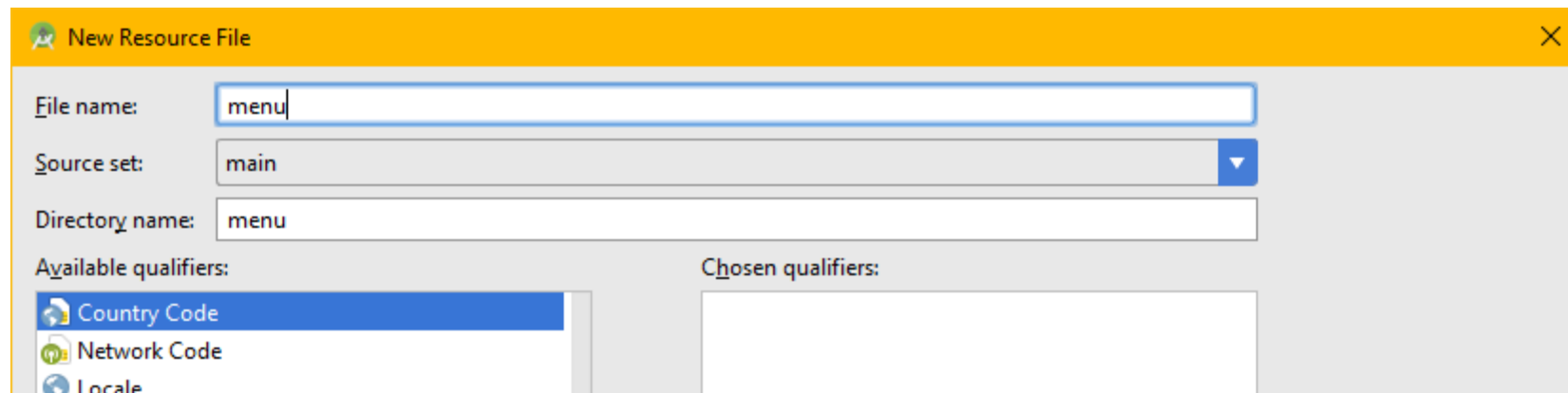
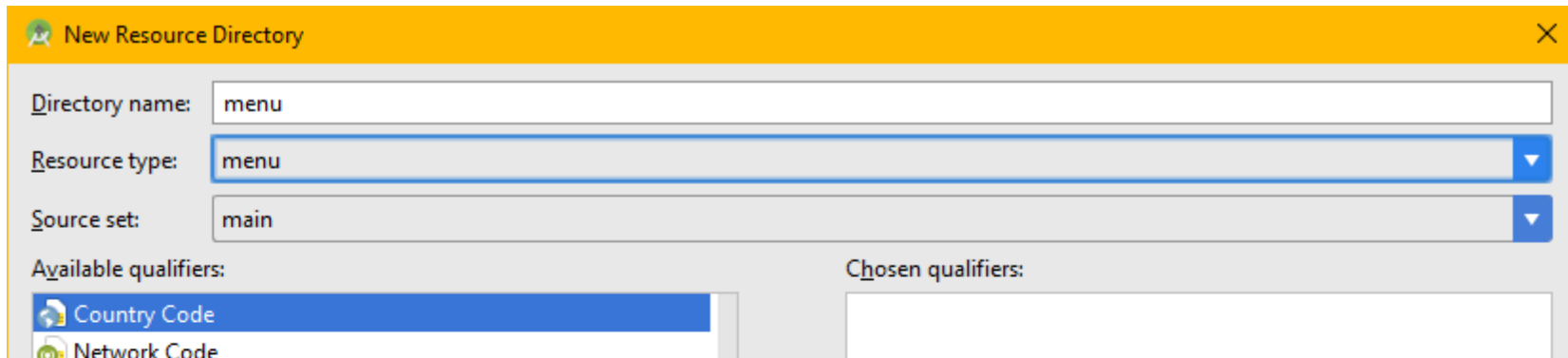
OK Cancel Help

Меню

- Создать .xml файл для описания элементов меню
- Установить атрибуты
- Инициализировать меню в .java файле с помощью `OnCreateOptionsMenu`
- Обработать нажатия на элементы меню с помощью `OnItemClicked`

Меню

1. Создать .xml файл для описания элементов меню



Меню

2. Установить атрибуты элементов меню

- android:id
- android:title — текст пункта меню
- app:showAsAction — never:



ifRoom:



- android:orderInCategory — порядок элементов меню

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <menu xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:app="http://schemas.android.com/apk/res-auto">
4     <item
5         android:id="@+id/action_settings"
6         android:title="Settings"
7         app:showAsAction="never"
8         android:orderInCategory="1"></item>
9 </menu>
```

Меню

3. Инициализировать меню в .java файле с помощью onCreateOptionsMenu

```
@Override  
public boolean onCreateOptionsMenu(Menu menu) {  
    getMenuInflater().inflate(R.menu.menu, menu);  
    return super.onCreateOptionsMenu(menu);  
}
```

4. Обработать нажатия на элементы меню с помощью onOptionsItemSelected

```
@Override  
public boolean onOptionsItemSelected(MenuItem item) {  
    int item_id = item.getItemId();  
    if (item_id == R.id.action_settings) {  
        Toast.makeText(this, "Setting menu item was clicked", Toast.LENGTH_LONG).show();  
        return true;  
    }  
  
    return super.onOptionsItemSelected(item);  
}
```

Построение запроса Uri (Uniform Resource Identifier)

```
Uri builtUri = Uri.parse(BASE_URL).buildUpon()  
    .appendQueryParameter(PARAM_NAME, PARAM_VALUE)  
    ..  
    .build();
```

Пример:

<https://yandex.ru/search/?lr=213&text=android> =>

```
Uri yandexUri =  
    Uri.parse(«https://yandex.ru/search/»).buildUpon()  
        .appendQueryParameter(«lr», «213»)  
        .appendQueryParameter(«text», «android»)  
        .build();
```


Построение запроса Url (Uniform Resource Locator)

```
URL url = null;
try {
    url = new URL(builtUri.toString());
} catch (MalformedURLException e) {
    e.printStackTrace();
}
```

Получение результата запроса из интернета

```
89  @
90      public static String getResponseFromHttpUrl(URL url) throws IOException {
91          HttpURLConnection urlConnection = (HttpURLConnection) url.openConnection();
92          try {
93              InputStream in = urlConnection.getInputStream();
94
95              Scanner scanner = new Scanner(in);
96              scanner.useDelimiter("\\A");
97
98              boolean hasInput = scanner.hasNext();
99              if (hasInput) {
100                  return scanner.next();
101              } else {
102                  return null;
103              }
104          } finally {
105              urlConnection.disconnect();
106          }
```

Permissions

- Каждое приложение на андроид работает в своем собственном экземпляре (instance) среды выполнения (runtime)
 - чтобы ни одно приложение не могло выполнять действия, которые могут повлиять на другие приложения, систему или пользователя
- => доступ к интернету, контактам, местоположению, .. требует разрешения
- Нужно выбирать минимум разрешений
- Добавление разрешения в AndroidManifest.xml

```
<uses-permission android:name="android.permission.INTERNET"/>
```

AsyncTask

- `AsyncTask<Params, Progress, Result>`
 - Generic-класс с тремя параметрами
 - Позволяет посылать сообщения из фонового потока для отображения текущего состояния в главном потоке
- Параметры
 - `Params`: тип параметров, которые посылаются задаче для выполнения в виде `Params... params`
 - `Progress`: тип для обновления текущего состояния выполнения задачи
 - `Result`: тип результата выполнения задачи

AsyncTask

- Основные методы:
 - OnPreExecute
 - doInBackground
 - onProgressUpdate
 - onPostExecute
- Для запуска асинхронной задачи, у объекта класса, реализующего AsyncTask вызывается метод **execute** с параметрами типа Params

AsyncTask

```
public class requestInternet extends AsyncTask<URL, Void, String> {

    @Override
    protected void onPreExecute() {
        //show ProgressBar
        //hide view where response will be shown
        super.onPreExecute();
    }

    @Override
    protected String doInBackground(URL... urls) {
        //get response from url
        return null;
    }

    @Override
    protected void onPostExecute(String s) {
        super.onPostExecute(s);
        //hide ProgressBar
        //show view where response will be shown
    }
}

new requestInternet().execute(searchURL);
```

JSON

Java Script Object Notation

- Формат, используемый для передачи данных
- Особенности:
 - Визуально понятен
 - Компактен
 - Легок в написании
 - Позволяет объявлять массивы
- Представляется в виде набора пар ключ-значение
 - Ключ — строка
 - Значение — строка / число / bool / null / другой набор пар / массив наборов пар с одинаковыми ключами

JSON / XML

Пример в JSON

```
1  {"menu": {  
2    "id": "file",  
3    "value": "File",  
4    "popup": {  
5      "menuitem": [  
6        {"value": "New", "onclick": "CreateNewDoc()"},  
7        {"value": "Open", "onclick": "OpenDoc()"},  
8        {"value": "Close", "onclick": "CloseDoc()"}  
9      ]  
10   }  
11 }}
```

Пример в XML

```
1  <menu id="file" value="File">  
2    <popup>  
3      <menuitem value="New" onclick="CreateNewDoc()" />  
4      <menuitem value="Open" onclick="OpenDoc()" />  
5      <menuitem value="Close" onclick="CloseDoc()" />  
6    </popup>  
7  </menu>
```


JSON

Чтение JSON-данных

```
1  {"menu": {  
2    "id": "file",  
3    "value": "File",  
4    "popup": {  
5      "menuitem": [  
6        {"value": "New", "onclick": "CreateNewDoc()"},  
7        {"value": "Open", "onclick": "OpenDoc()"},  
8        {"value": "Close", "onclick": "CloseDoc()"}  
9      ]  
10   }  
11  }}
```

```
JSONObject menuObj = new JSONObject(JsonString);  
JSONObject menu = menuObj.getJSONObject("menu");
```

```
String id = menu.getString("id");
```

```
JSONObject popup = menuObj.getJSONObject("popup");  
JSONArray menuItem = popup.getJSONArray("menuitem");  
String value0 = menuItem.getJSONObject(0).getString("value");
```

Лабораторная работа 2.

- 1) Создать приложение для 2 ориентаций экрана, 2 языков
На примере рисунка и текстового поля
- 2) Добавить пункт меню «Обновить» в ActionBar, пункт меню «Поменять язык»
- 3) Выгрузить json - данные со страницы
например: <https://newsapi.org/> , <https://openweathermap.org/current>
- 3*. Сделать выпадающее меню с выбором города для погоды / ресурса для новостей
- 4) Достать нужные поля с помощью библиотеки Json
- 5) Вставить данные в TextView
- 5*. Вставить данные в TableLayout