

CV 2

ListView, Adapter

https://github.com/codepath/android_guides/wiki/Using-a-BaseAdapter-with-ListView

- ListView
- BaseAdapter
- model
- ViewHolder
- RecyclerView - lepší, rychlejší, ale složitější

Networking

- obrázky - Picasso & Fresco
- JSON (GSON)
- HttpURLConnection
- OkHttpClient
- Retrofit 2

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

Obrázky - Picasso

<https://github.com/square/picasso>

```
compile 'com.squareup.picasso:picasso:2.5.2'
```

```
Picasso.with(context)  
    .load("http://...")  
    .into(imageView)
```

Zadání

- Připravte si Picasso
- V ListView zobrazte i obrázek ke jménu
- Velikost obrázku 50x50

Řešení

```
<ImageView  
    android:id="@+id/image"  
    ... />
```

```
ImageView image = (ImageView) findViewById(R.id.image);
```

Picasso

```
.with(this)  
.load("...")  
.into(image);
```

Obrázky - Fresco

<http://frescolib.org/>

- Facebook
- ashmem vs. OutOfMemory
- ProgressiveJPEG

```
<com.facebook.drawee.view.SimpleDraweeView  
    ...  
>
```

```
Uri uri = Uri.parse("...");
```

```
SimpleDraweeView view = (SimpleDraweeView) findViewById(R.id.image);
```

```
view.setImageURI(uri);
```

JSON - GSON

<https://github.com/google/gson>

- GSON je knihovna pro zpracování JSON do objektu a naopak

JSON - GSON

```
compile 'com.google.code.gson:gson:2.2.4'
```

```
Gson gson = new Gson();  
User user = new User();  
String json = gson.toJson(user);  
User user2 = gson.fromJson(json, User.class);
```

```
public class User {  
    public String name;  
}
```

Zadání

- Vytvořte třídu Teacher, která bude mít 2 Stringy - name a surname.
- Inicializujte s hodnotami a převed'te do JSONu pomocí GSONu
- Výsledek zobrazte jako Toast

Řešení

```
public class Teacher {  
    public String name;  
    public String surname;  
}
```

```
Teacher t = new Teacher();  
t.name = "Radim";  
t.surname = "Vaculik";
```

```
String json = new Gson().toJson(t);
```

```
Toast.makeText(this, json, Toast.LENGTH_LONG).show();
```

HttpURLConnection

<https://developer.android.com/reference/java/net/HttpURLConnection.html>

```
URL url = new URL("http://.../");
HttpURLConnection urlConnection =
(HttpURLConnection) url.openConnection();
try {
    InputStream in = new
BufferedInputStream(urlConnection.getInputStream());
    readStream(in);
} finally {
    urlConnection.disconnect();
}
```

OkHttpClient - GET

<http://square.github.io/okhttp/>

```
OkHttpClient client = new OkHttpClient();

Request request = new Request.Builder()
    .url(url)
    .build();

Response response = client
    .newCall(request)
    .execute();

return response.body().string();
```

OkHttpClient - POST

<http://square.github.io/okhttp/>

```
OkHttpClient client = new OkHttpClient();

Request request = new Request.Builder()
    .url(url)
    .post(RequestBody.create(..., json))
    .build();

Response response = client
    .newCall(request)
    .execute();

return response.body().string();
```

Retrofit 2

<https://square.github.io/retrofit/>

- knihovna pro HTTP requests
- Request, Response, Call
- Synchronní a asynchronní

Retrofit 2

<https://square.github.io/retrofit/>

```
public interface GitHubService {
    @GET("users/{user}/repos")
    Call<List<Repo>> listRepos(@Path("user") String user);
}

Retrofit retrofit = new Retrofit.Builder()
    .baseUrl("https://api.github.com/")
    .build();

GitHubService service = retrofit.create(GitHubService.class);

Call<List<Repo>> repos = service.listRepos("octocat");
```


Zadání

<https://developer.github.com/v3/users/>

- Aplikace, která zobrazí fotku uživatele z Githubu
- <https://api.github.com/users/octocat>
- avatar_url