

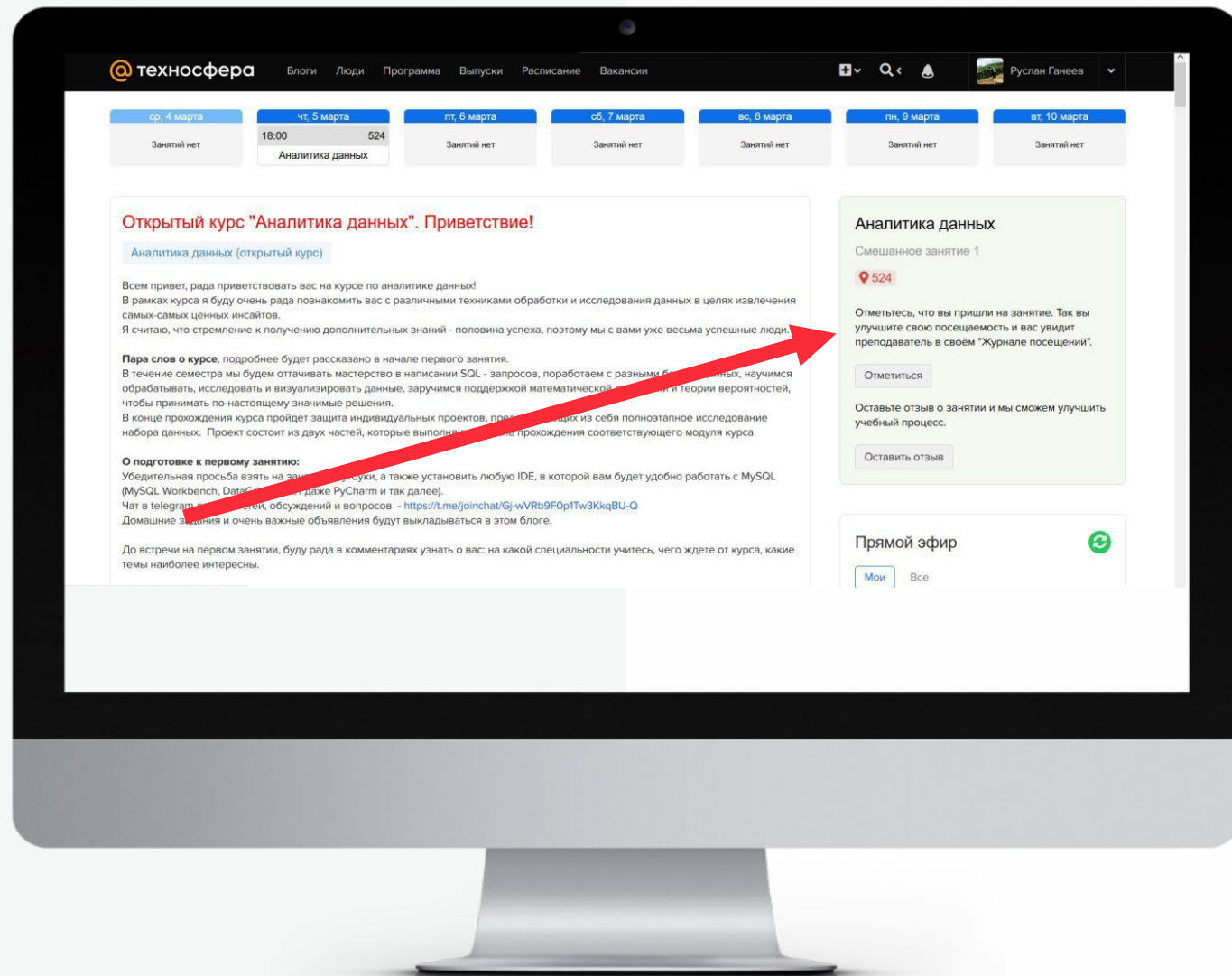


x @ mail.ru
group


Хранение данных

Куликов Никита





Напоминание
отметиться
на портале



Содержание занятия

1. Сеть
2. Хранилище

Сеть

1. OkHttp
2. Retrofit

OkHttp

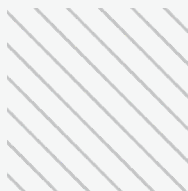
```
OkHttpClient client = new OkHttpClient();

public String loadByUrl(String url) throws IOException {
    Request request = new Request.Builder()
        .url(url)
        .build();

    try (Response response = client.newCall(request).execute()) {
        return response.body().string();
    }
}
```



Моки



Просто строка

```
public String loadByUrl(String url) throws IOException {  
    return "My str"  
}
```

designer.mocky.io

Your mock is ready!

Mock URL 

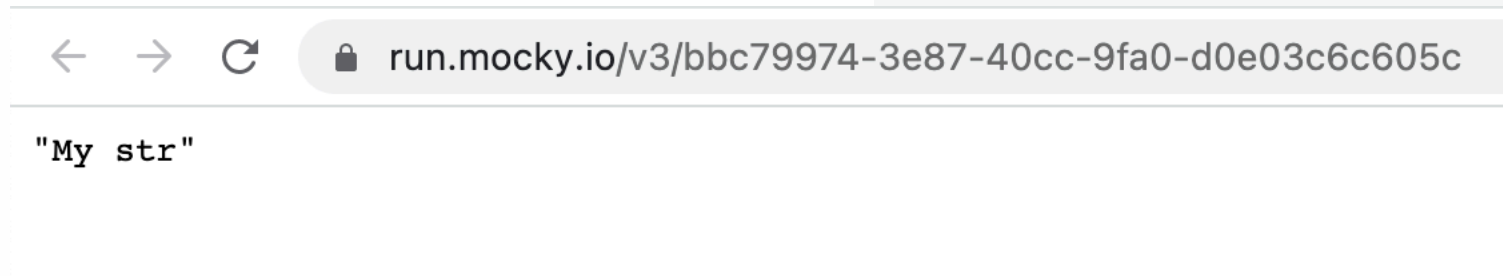
<https://run.mocky.io/v3/bbc79974-3e87-40cc-9fa0-d0e03c6c605c>

Secret delete link 

<https://designer.mocky.io/manage/delete/bbc79974-3e87-40cc-9fa0-d0e03c6c605c/0Xo0Y9n1fkpYvTiyZrshVIUa04xLLs40fJhH>

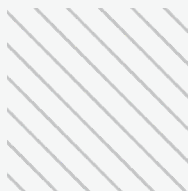


designer.mocky.io





Retrofit



Как пользоваться?

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

```
public void load() throws IOException {  
    Retrofit retrofit = new Retrofit.Builder()  
        .baseUrl("https://api.github.com/")  
        .build();  
  
    GitHubService service = retrofit.create(GitHubService.class);  
    service.listRepos("lionzxy").execute();  
}
```

Почитаем javadoc

```
/**  
 * Synchronously send the request and return its response.  
 *  
 * @throws IOException if a problem occurred talking to the server.  
 * @throws RuntimeException (and subclasses) if an unexpected error occurs creating the request  
 * or decoding the response.  
 */  
Response<T> execute() throws IOException;
```

Async call

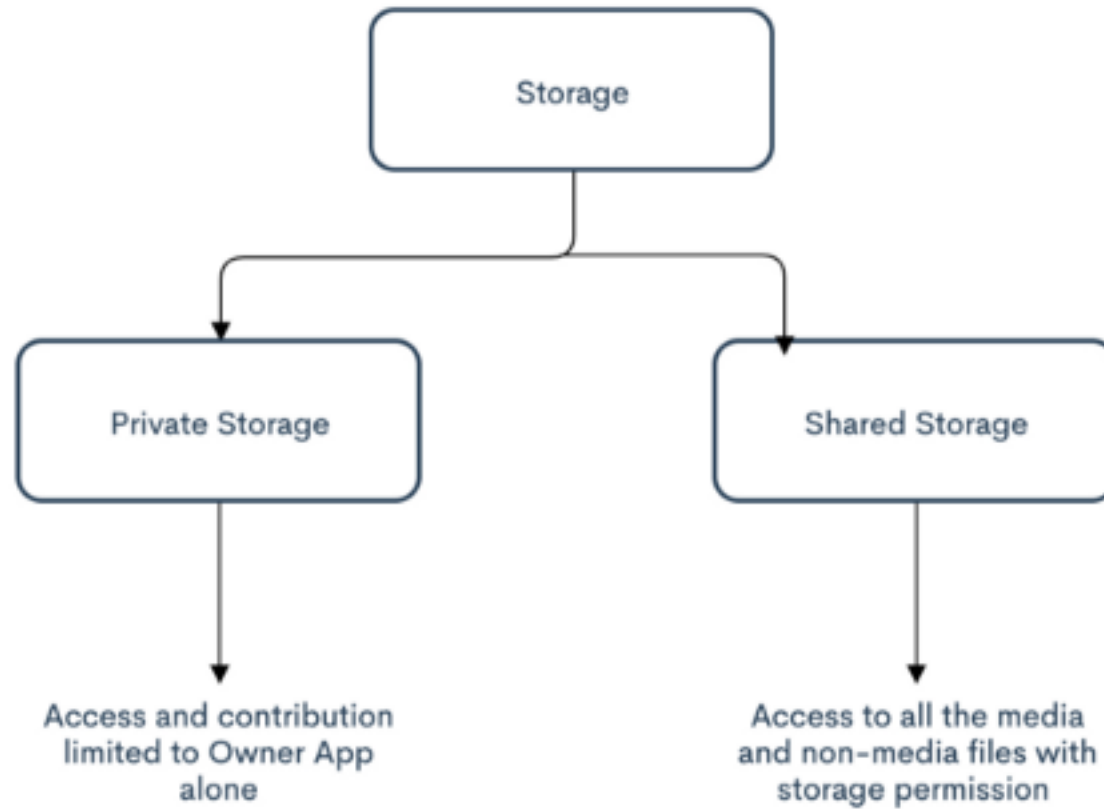
```
service.listRepos("lionzxy").enqueue(new Callback<List<String>>() {  
    @Override  
    public void onResponse(Call<List<String>> call, retrofit2.Response<List<String>> response) {  
  
    }  
  
    @Override  
    public void onFailure(Call<List<String>> call, Throwable t) {  
  
    }  
});
```



Storage



Хранилище в Android

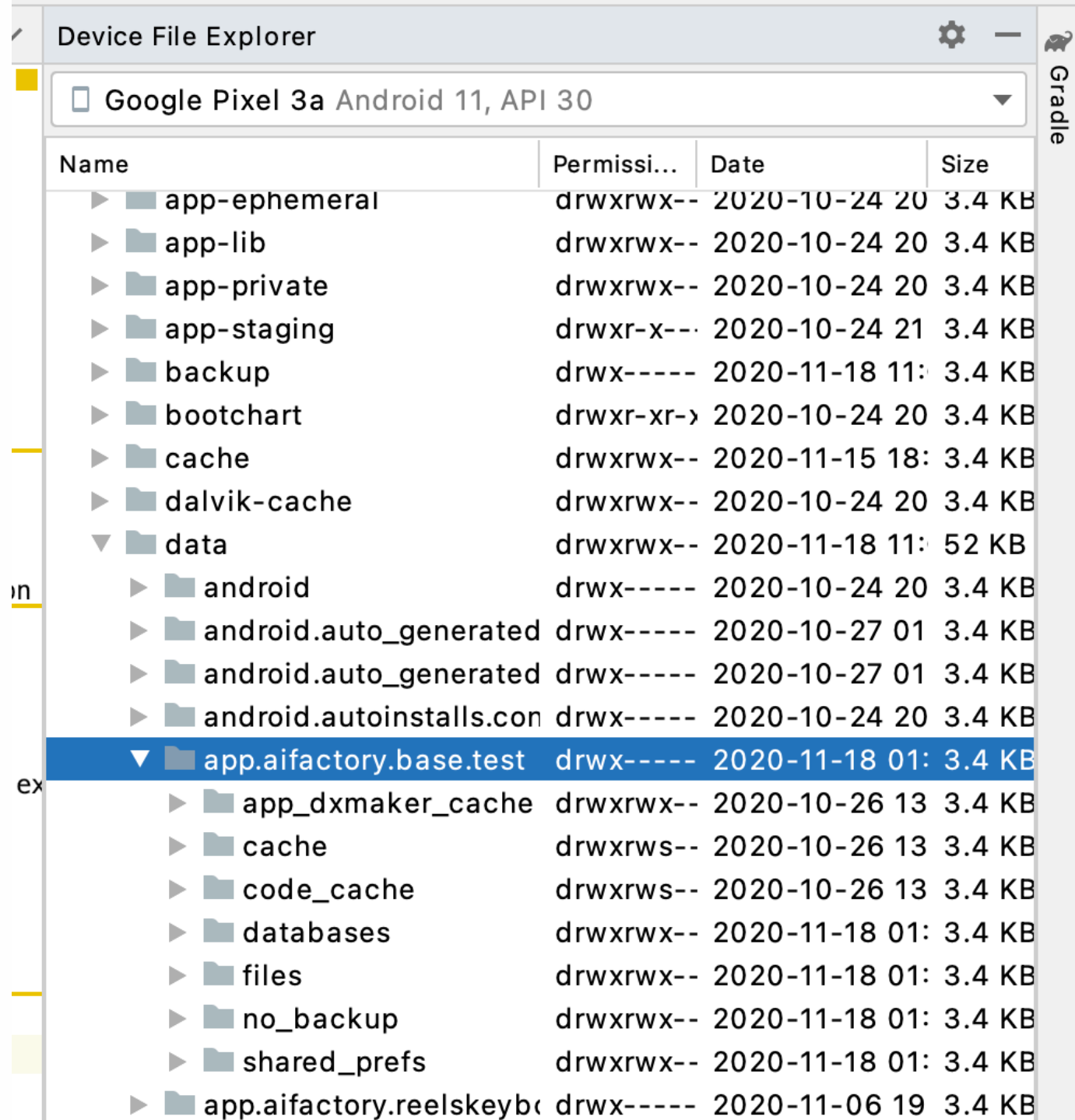


Internal storage

- `getCacheDir()`
- `getDir()`
- `getDatabasePath()`
- `getFilesDir()`
- `openFileInput()`
- `openFileOutput()`

`/data/data/packageName/`

#016



Device File Explorer

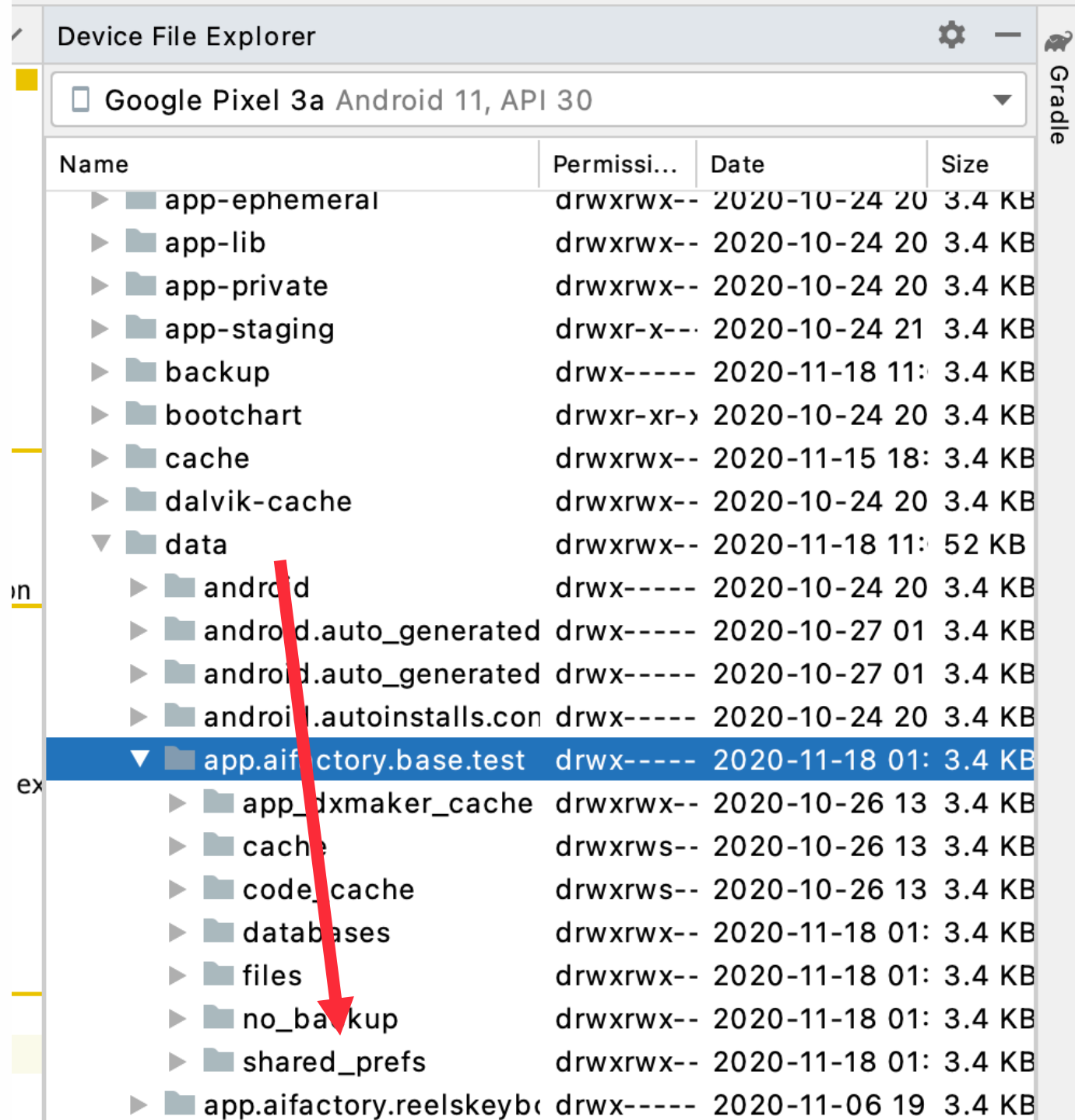
Google Pixel 3a Android 11, API 30

Name	Permissi...	Date	Size
▶ app-ephemeral	drwxrwx--	2020-10-24 20	3.4 KB
▶ app-lib	drwxrwx--	2020-10-24 20	3.4 KB
▶ app-private	drwxrwx--	2020-10-24 20	3.4 KB
▶ app-staging	drwxr-x---	2020-10-24 21	3.4 KB
▶ backup	drwx-----	2020-11-18 11:	3.4 KB
▶ bootchart	drwxr-xr->	2020-10-24 20	3.4 KB
▶ cache	drwxrwx--	2020-11-15 18:	3.4 KB
▶ dalvik-cache	drwxrwx--	2020-10-24 20	3.4 KB
▼ data	drwxrwx--	2020-11-18 11:	52 KB
▶ android	drwx-----	2020-10-24 20	3.4 KB
▶ android.auto_generated	drwx-----	2020-10-27 01	3.4 KB
▶ android.auto_generated	drwx-----	2020-10-27 01	3.4 KB
▶ android.autoinstalls.con	drwx-----	2020-10-24 20	3.4 KB
▼ app.aifactory.base.test	drwx-----	2020-11-18 01:	3.4 KB
▶ app_dxmaker_cache	drwxrwx--	2020-10-26 13	3.4 KB
▶ cache	drwxrws--	2020-10-26 13	3.4 KB
▶ code_cache	drwxrws--	2020-10-26 13	3.4 KB
▶ databases	drwxrwx--	2020-11-18 01:	3.4 KB
▶ files	drwxrwx--	2020-11-18 01:	3.4 KB
▶ no_backup	drwxrwx--	2020-11-18 01:	3.4 KB
▶ shared_prefs	drwxrwx--	2020-11-18 01:	3.4 KB
▶ app.aifactory.reelskeybo	drwx-----	2020-11-06 19	3.4 KB

Internal storage

- `getCacheDir()`
- `getDir()`
- `getDatabasePath()`
- `getFilesDir()`
- `openFileInput()`
- `openFileOutput()`

#017



Device File Explorer

Google Pixel 3a Android 11, API 30

Name	Permissi...	Date	Size
▶ app-ephemeral	drwxrwx--	2020-10-24 20	3.4 KB
▶ app-lib	drwxrwx--	2020-10-24 20	3.4 KB
▶ app-private	drwxrwx--	2020-10-24 20	3.4 KB
▶ app-staging	drwxr-x---	2020-10-24 21	3.4 KB
▶ backup	drwx-----	2020-11-18 11:	3.4 KB
▶ bootchart	drwxr-xr->	2020-10-24 20	3.4 KB
▶ cache	drwxrwx--	2020-11-15 18:	3.4 KB
▶ dalvik-cache	drwxrwx--	2020-10-24 20	3.4 KB
▼ data	drwxrwx--	2020-11-18 11:	52 KB
▶ android	drwx-----	2020-10-24 20	3.4 KB
▶ android.auto_generated	drwx-----	2020-10-27 01	3.4 KB
▶ android.auto_generated	drwx-----	2020-10-27 01	3.4 KB
▶ android.autoinstalls.con	drwx-----	2020-10-24 20	3.4 KB
▼ app.aifactory.base.test	drwx-----	2020-11-18 01:	3.4 KB
▶ app_dxmaker_cache	drwxrwx--	2020-10-26 13	3.4 KB
▶ cache	drwxrws--	2020-10-26 13	3.4 KB
▶ code_cache	drwxrws--	2020-10-26 13	3.4 KB
▶ databases	drwxrwx--	2020-11-18 01:	3.4 KB
▶ files	drwxrwx--	2020-11-18 01:	3.4 KB
▶ no_backup	drwxrwx--	2020-11-18 01:	3.4 KB
▶ shared_prefs	drwxrwx--	2020-11-18 01:	3.4 KB
▶ app.aifactory.reelskeybo	drwx-----	2020-11-06 19	3.4 KB

SharedPreferences

```
public static final String PREFS_NAME = "MyPrefsFile";

public void readPref() {
    SharedPreferences settings
        = getSharedPreferences(PREFS_NAME, 0);
    boolean silent = settings.getBoolean("silentMode", false);
}

public void writePref() {
    SharedPreferences settings = getSharedPreferences(PREFS_NAME, 0);
    SharedPreferences.Editor editor = settings.edit();
    editor.putBoolean("silentMode", mSilentMode);
    editor.apply();
}
```

External storage

Добавить пермишены в `AndroidManifest.xml`:

- `WRITE_EXTERNAL_STORAGE`
- `READ_EXTERNAL_STORAGE`

Проверка состояния внешнего хранилища

`Environment.getExternalStorageState()`

Возможные состояния хранилища:

- `Environment.MEDIA_MOUNTED`
- `Environment.MEDIA_MOUNTED_READ_ONLY`

External storage

```
public boolean isExternalStorageWritable() {  
    String state = Environment.getExternalStorageState();  
    return Environment.MEDIA_MOUNTED.equals(state);  
}  
  
public boolean isExternalStorageReadable() {  
    String state = Environment.getExternalStorageState();  
    return Environment.MEDIA_MOUNTED.equals(state) ||  
        Environment.MEDIA_MOUNTED_READ_ONLY.equals(state);  
}
```

“Общедоступные” директории

Android сканирует некоторые директории с целью предоставить пользователю удобный доступ к ним. Получить путь до них можно с помощью:

```
File getExternalStoragePublicDirectory  
                                     (String type)
```

Типы директорий:

- DIRECTORY_MUSIC
- DIRECTORY_PICTURES
- DIRECTORY_DOWNLOADS
- DIRECTORY_RINGTONES
- ...

Приватные внешние директории

Как Internal Storage для вашего приложения, но на внешнем носителе:

- `File getExternalFilesDir(String type)` – путь до приватной директории приложения
- `File getExternalCacheDir()` – путь до директории для хранения кэшей

```
<uses-permission
```

```
    android:name="android.permission.WRITE_EXTERNAL_STORAGE"
```

```
    android:maxSdkVersion="18"
```

```
/>
```

Куда останавливается приложение

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"  
    android:installLocation="preferExternal"  
    ... >
```

Возможные варианты для `installLocation`:

- `internalOnly`
- `auto`
- `preferExternal`



База данных



Room - Entity

```
@Entity(tableName = "word_table")
public class Word {

    @PrimaryKey
    @NonNull
    @ColumnInfo(name = "word")
    private String mWord;

    public Word(@NonNull String word) {
        this.mWord = word;
    }

    public String getWord() {
        return this.mWord;
    }
}
```

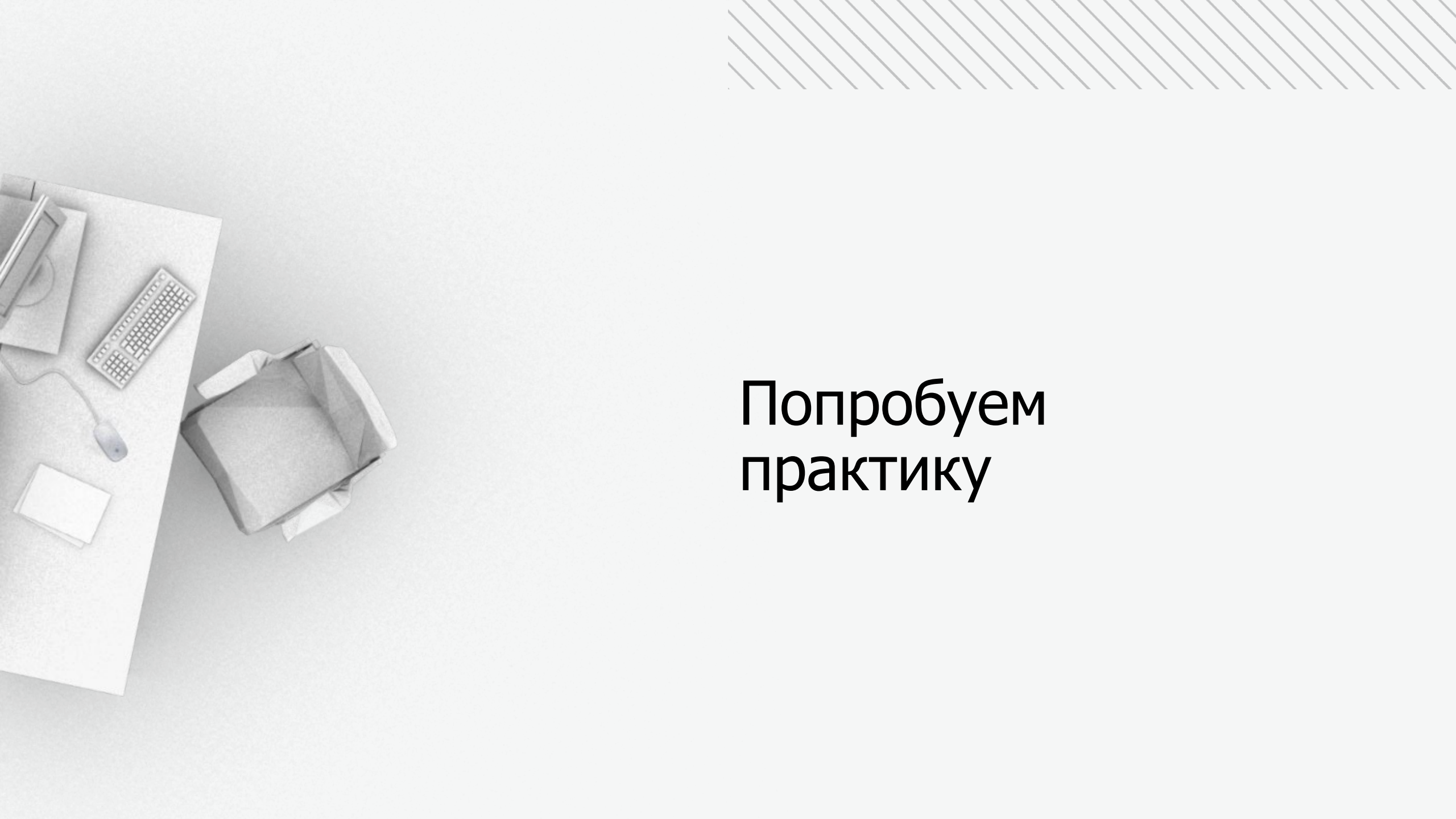
Room - DAO

```
@Dao
public interface WordDao {

    // allowing the insert of the same word multiple times by passing a
    // conflict resolution strategy
    @Insert(onConflict = OnConflictStrategy.IGNORE)
    void insert(Word word);

    @Query("DELETE FROM word_table")
    void deleteAll();

    @Query("SELECT * FROM word_table ORDER BY word ASC")
    List<Word> getAlphabetizedWords();
}
```



Попробуем
практику

**СПАСИБО
ЗА ВНИМАНИЕ**

