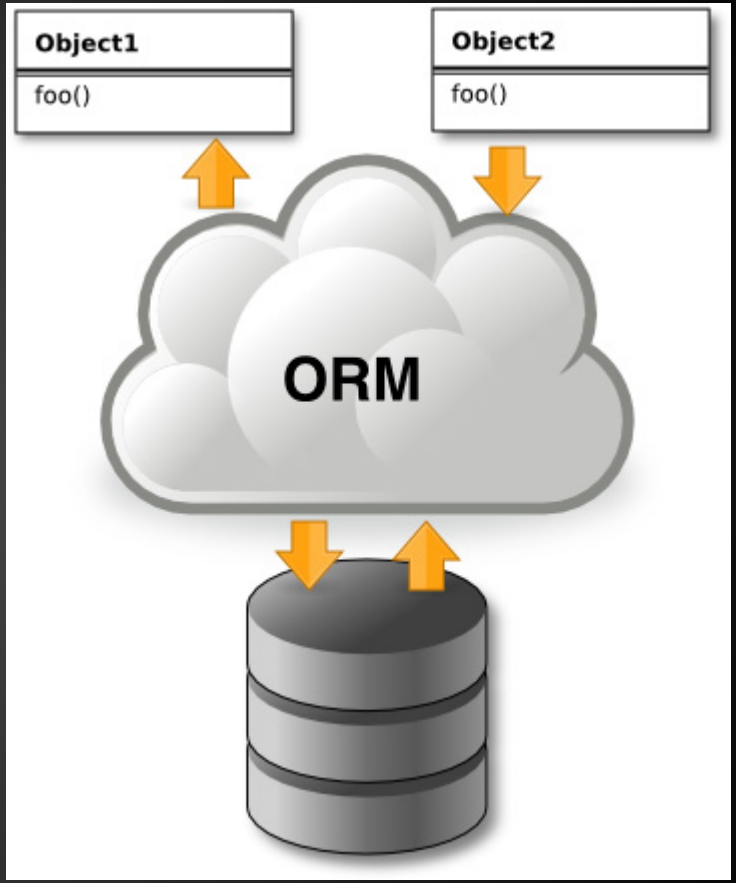
# Android Fundamentals

Лекция 5

# Storing Data using ORM

# ORM - object-relational mapping



# ORMLite

- Open source
- Well-documented
- Supports loaders and cursor adapters
- Works not only with Android SQLite
- A lot of discussions on SOF

# Object annotation

```
public class Person {

    public static final String COLUMN_NAME_NAME = "name";

    @DatabaseField(generatedId = true, columnName = "_id")
    private long id;

    @DatabaseField(columnName = COLUMN_NAME_NAME, uniqueCombo = true)
    private String name;

    @DatabaseField(uniqueCombo = true)
    private String lastName;

    @DatabaseField(foreign = true, foreignAutoRefresh = true)
    private Company company;
```

# OrmHelper

...

# Selection

```
QueryBuilder<Account, String> queryBuilder =
        accountDao.queryBuilder();

Where<Account, String> where = queryBuilder.where();
where.eq(Account.NAME_FIELD_NAME, "foo");
where.and();
where.eq(Account.PASSWORD_FIELD_NAME, "_secret");
PreparedQuery<Account> preparedQuery = queryBuilder.prepare();

List<Account> accounts = preparedQuery.query();
```

# Custom dao

```
public class CompanyDao extends CommonDao<Company, Long> {

    public static final String TAG = CompanyDao.class.getSimpleName();

    public CompanyDao(ConnectionSource connectionSource) { super(connectionSource, Compan

    public List<Company> searchByName(String name) throws SQLException {
        return queryBuilder().where().like(Company.COLUMN_NAME_NAME, name + "%").query();
    }

    public void create(Company company, List<Person> persons, CommonDao<Person, Long> per
        create(company);
        savePersons(persons, personDao);
    }
```

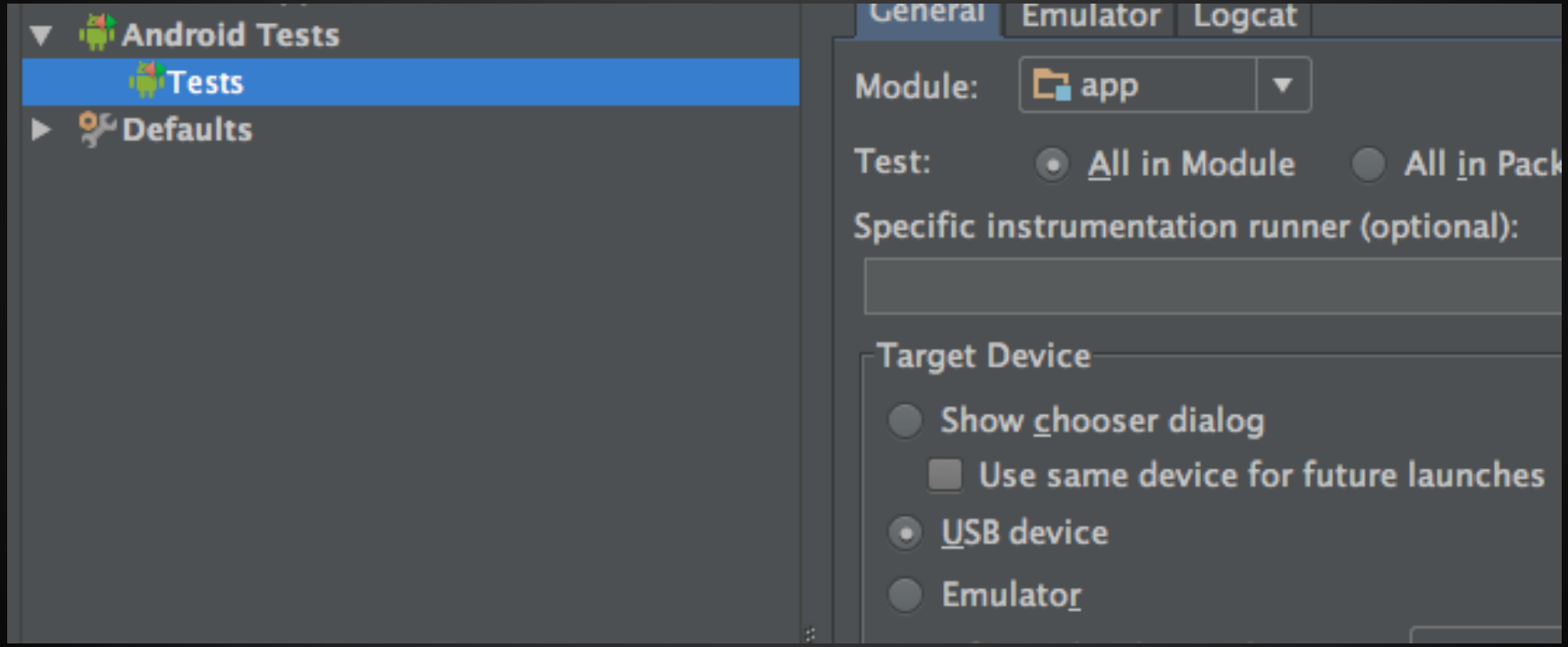
# OrmLiteCursorAdapter

```
CommonDao personDao = ormHelper.getDaoByClass(Person.
class);
PreparedQuery query = personDao.queryBuilder().prepare();
OrmCursorLoaderCallback<Person, Long> personLoaderCallback =
            new OrmCursorLoaderCallback<Person, Long>
(getActivity(), personDao, query, adapter);

getLoaderManager().initLoader(ORM_LOADER_ID, null,
personLoaderCallback);
```

# Tests in Android

# Unit Tests


Android Tests
Tests
Defaults
General
Emulator
Logcat
Module:
app
Test:
All in Module
All in Pack
Specific instrumentation runner (optional):
Target Device
Show chooser dialog
Use same device for future launches
USB device
Emulator

# Unit Tests

```
public class MyClassTest extends AndroidTestCase {
    @Override
    public void setUp() throws Exception {
        super.setUp();
        ...    }
    myFirstTest(){}
    @Override
    public void tearDown() throws Exception {
        super.tearDown();
        …     }
```

# UI tests

ESPRESSO 2.0

https://code.google.com/p/android-test-kit

# Unit tests on local jvm

Roboelectric

http://robolectric.org/

# Android studio native support

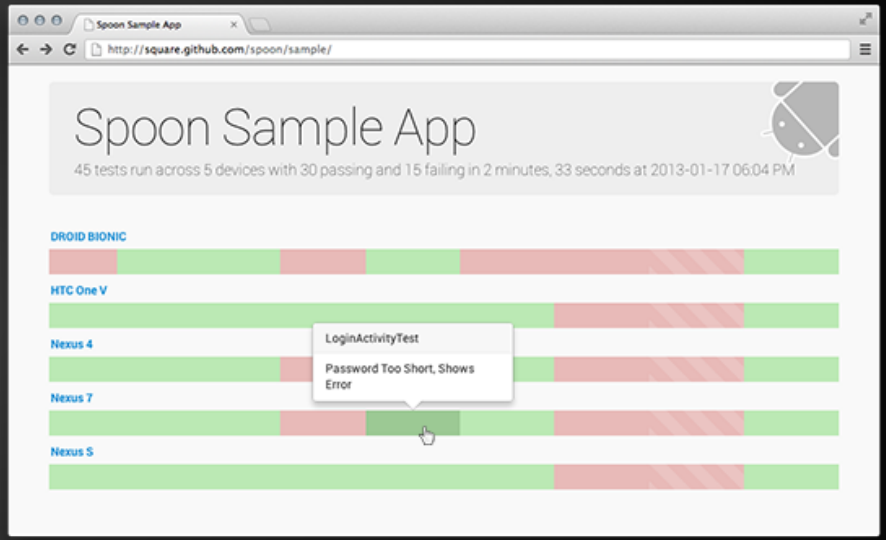
JUnit 4  + Mockito

https://sites.google.com/a/android.com/tools/tech-docs/unit-testing-support

# Multiply device test distribution

Spoon

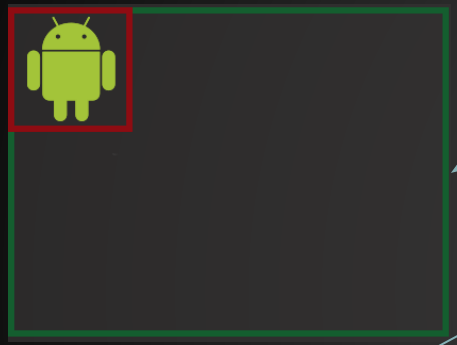
http://square.github.io/spoon/



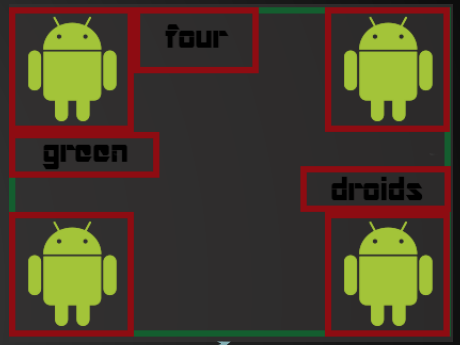
# IRL

# Best Practices for User Interface

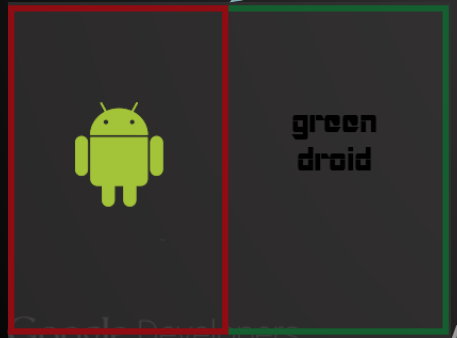
# View and ViewGroup. Layouts



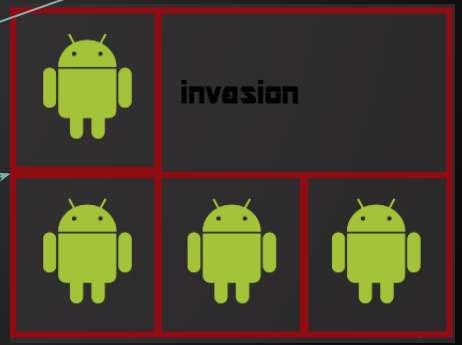
FrameLayout




Linear Layout




RelativeLayout




GridLayout

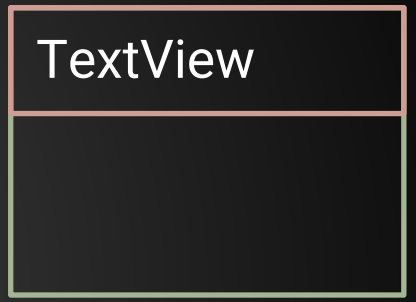
# View and ViewGroup

- Keep layout shallow and wide
- more siblings, fewer children
- Avoid excessive views

# View and ViewGroup. Width/height

width =
**wrap_content**
height =
**wrap_content**

width =
**match_parent**
height =
**wrap_content**



width =
**wrap_content**
height =
**match_parent**

width =
**match_parent**
height =
**match_parent**

# View and ViewGroup. Dimension

mm

Millimeters - Based on the physical size of the screen.

in

Inches - Based on the physical size of the screen.

pt

Points - 1/72 of an inch based on the physical size of the screen.

px

Pixels - Corresponds to actual pixels on the screen.

# View and ViewGroup. Dimension

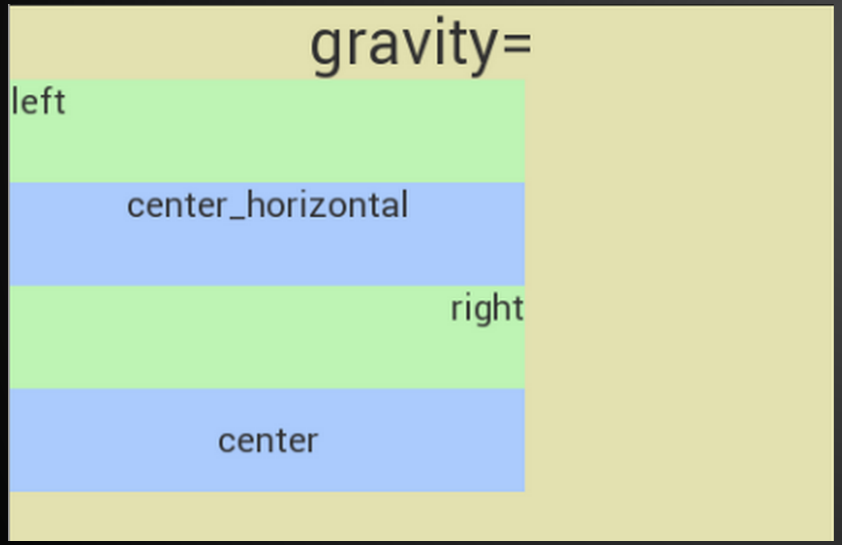
dip(dp)

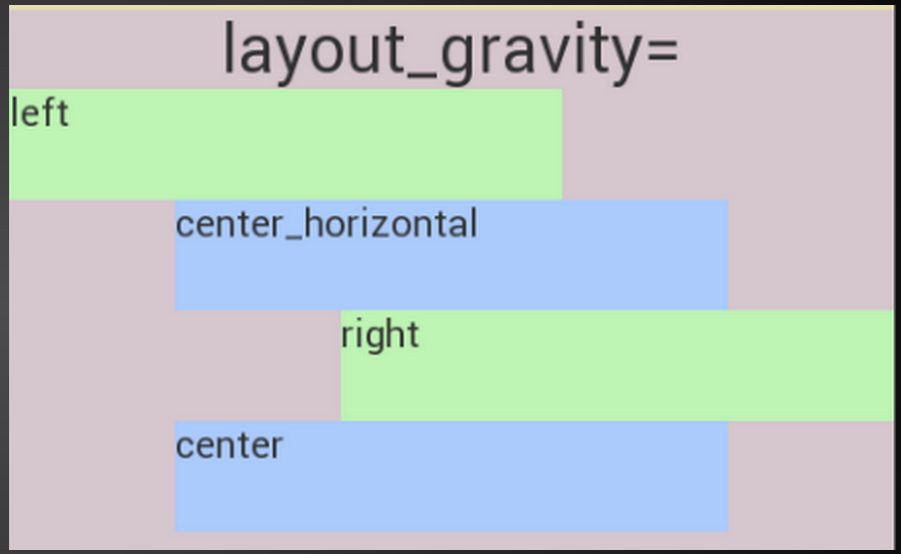
Density-independent Pixels - An abstract unit that is based on the physical density of the screen. These units are relative to a 160 dpi (dots per inch) screen, on which 1dp is roughly equal to 1px.

sp

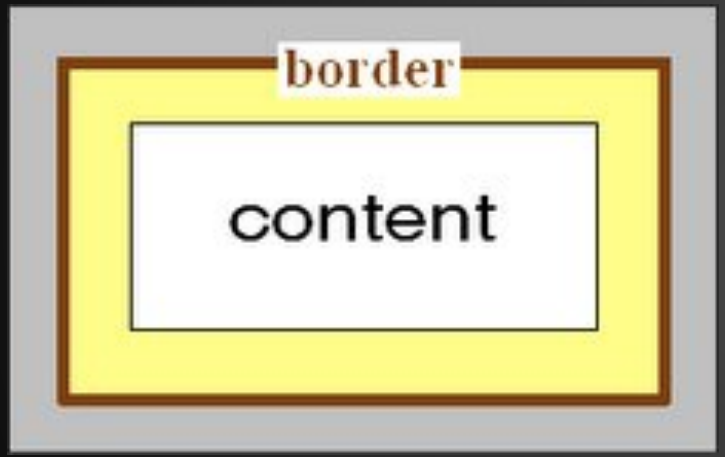
Scale-independent Pixels - This is like the dp unit, but it is also scaled by the user's font size preference. It is recommend you use this unit when specifying font sizes.

# View and ViewGroup. Gravity



# View and ViewGroup. Padding vs margin

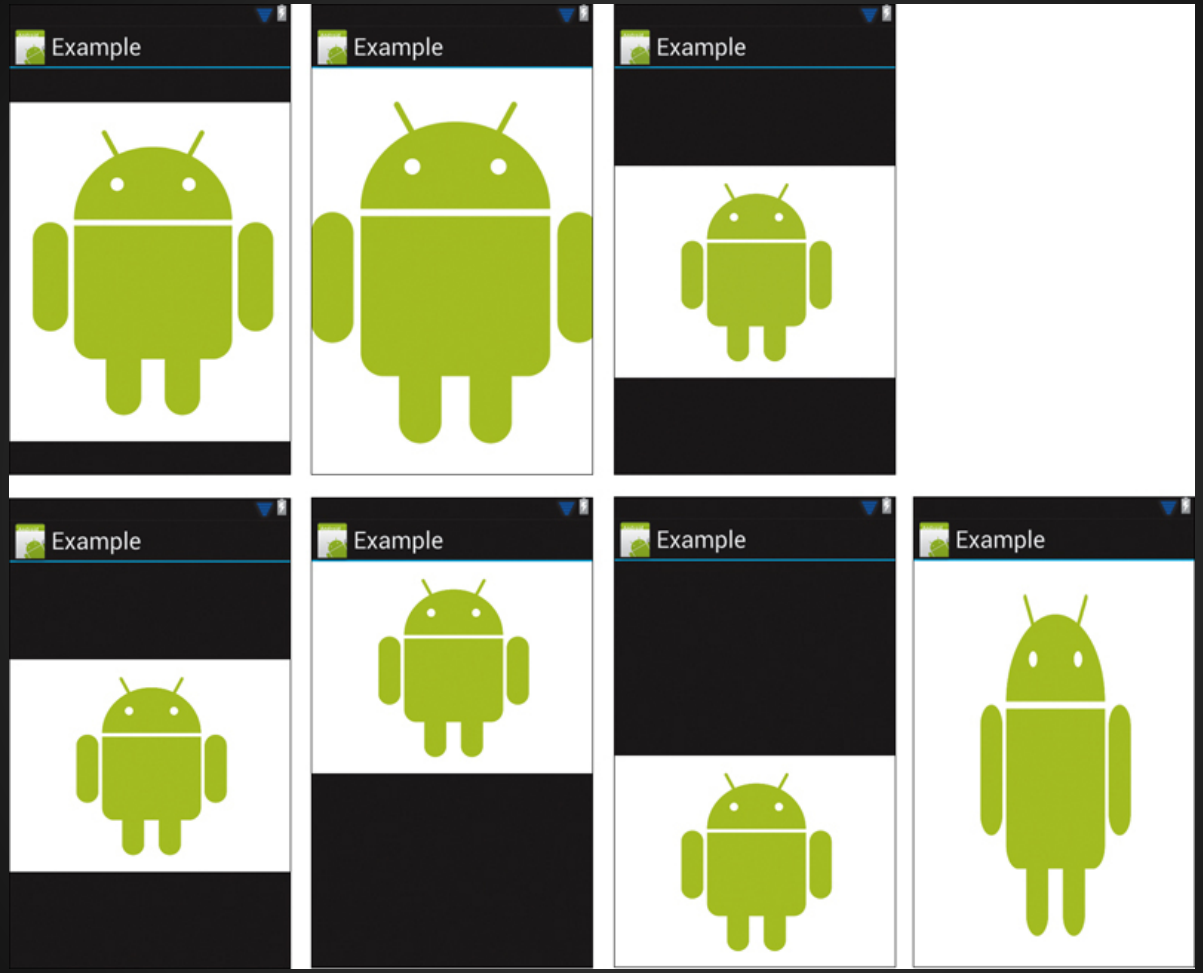



padding

margin

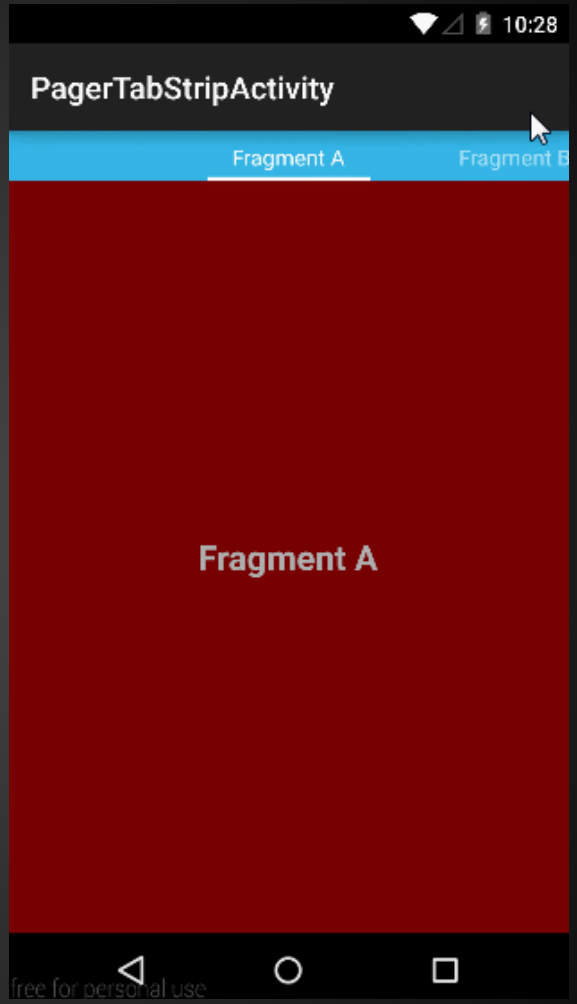
# View and ViewGroup. ScaleType

- CENTER
- CENTER_CROP
- CENTER_INSIDE
- FIT_CENTER
- FIT_END
- FIT_START
- FIT_XY
- MATRIX


Example
Example
Example
Example
Example
Example
Example
Example

# Effective Navigation

# Effective Navigation. ViewPager

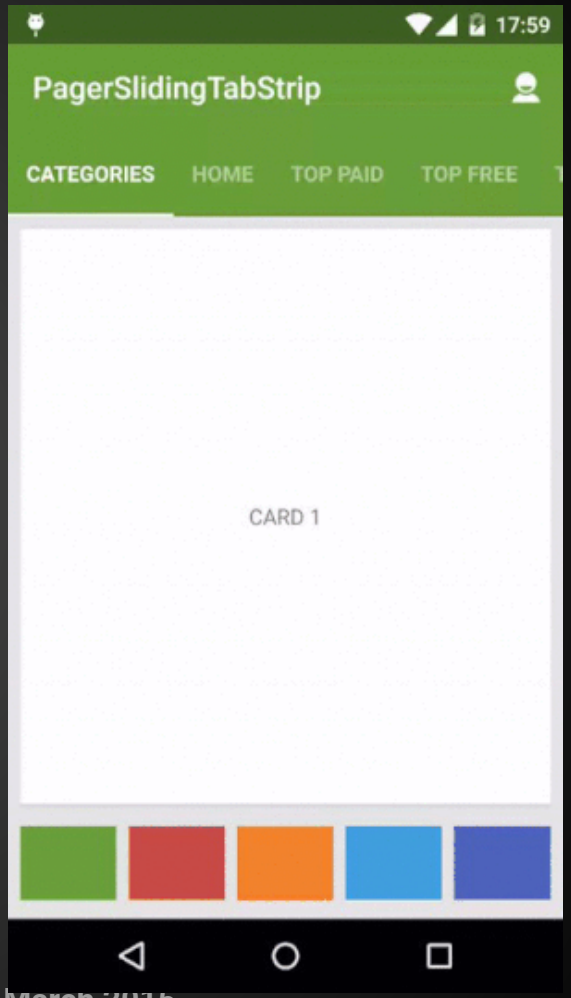



free for personal use

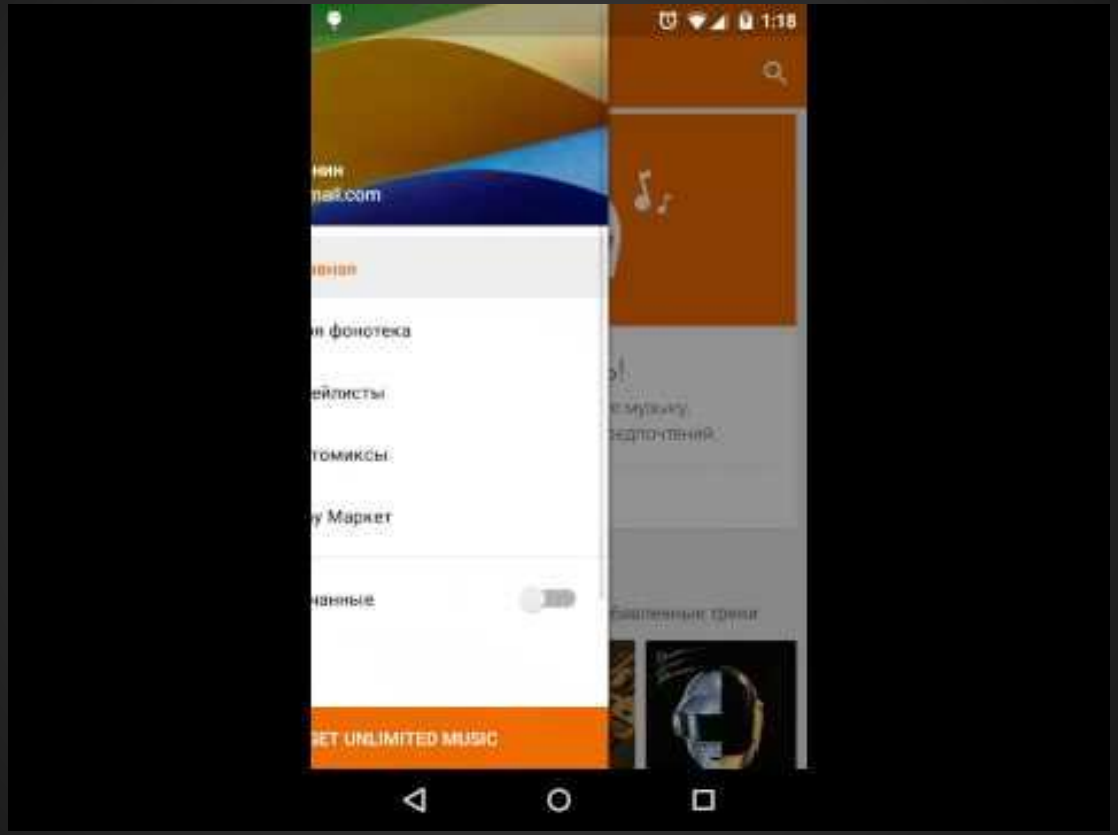
# Effective Navigation. ViewPagerIndicator

# Effective Navigation.

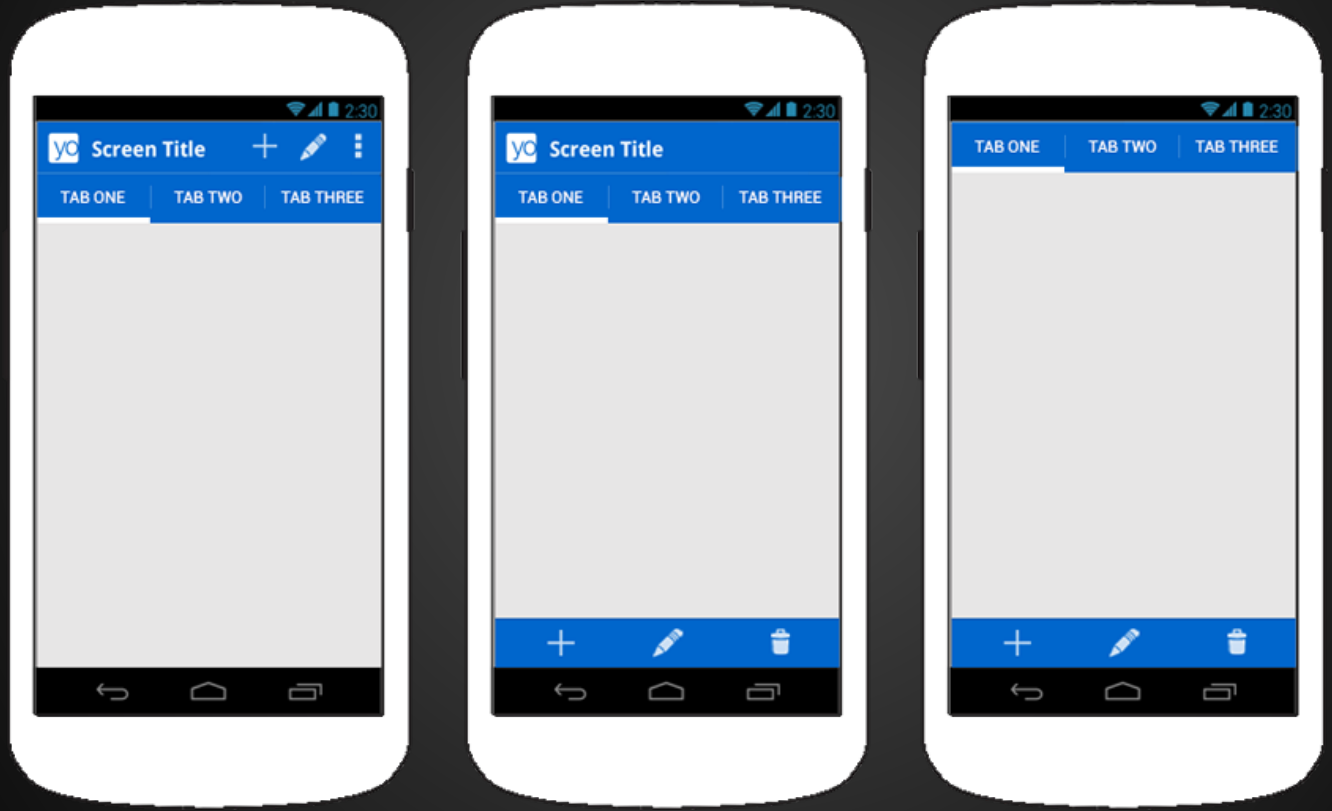
# PagerSlidingTabStrip



# Effective Navigation. Navigation Drawer



# Effective Navigation. Action Bar



# Effective Navigation. Menu

**res/menu/main_activity_menu.xml**

```
<menu xmlns:android="http://schemas.android.com/apk/res/android" >
    <item android:id="@+id/action_search"
        android:icon="@drawable/ic_action_search"
        android:title="@string/action_search"/>
    <item android:id="@+id/action_compose"
        android:icon="@drawable/ic_action_compose"
        android:title="@string/action_compose" />
</menu>
```

# Effective Navigation. Menu

```
@Override
public boolean onCreateOptionsMenu(Menu menu) {
    // Inflate the menu items for use in the action bar
    MenuInflater inflater = getMenuInflater();
    inflater.inflate(R.menu.main_activity_menu, menu);
    return super.onCreateOptionsMenu(menu);
}
```

# Effective Navigation. Menu

```
@Override
public boolean onOptionsItemSelected(MenuItem item) {
    switch (item.getItemId()) {
        case R.id.action_search:
            openSearch();
            return true;
        case R.id.action_compose:
            composeMessage();
            return true;
        default:
            return super.onOptionsItemSelected(item);
    }
}
```

# Effective Navigation. Menu

## android:showAsAction

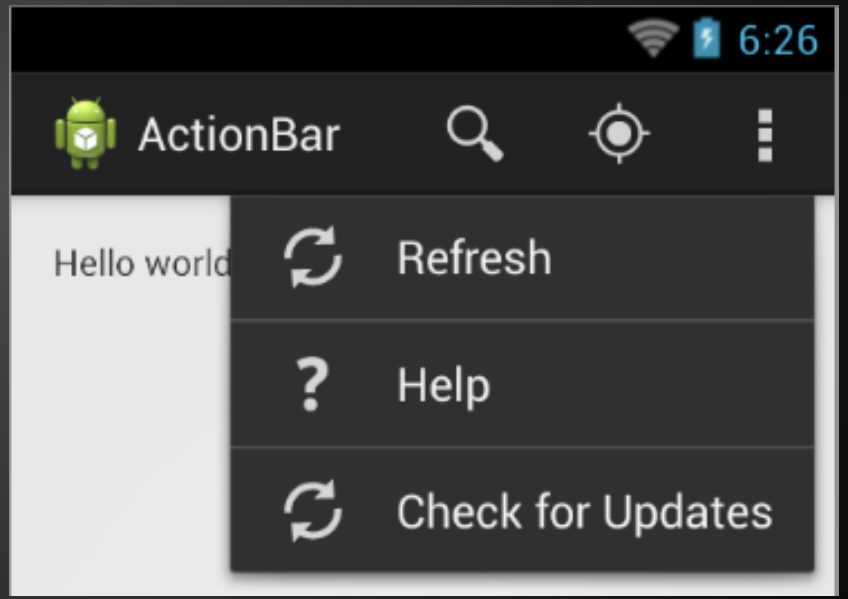
"ifRoom"

"never"

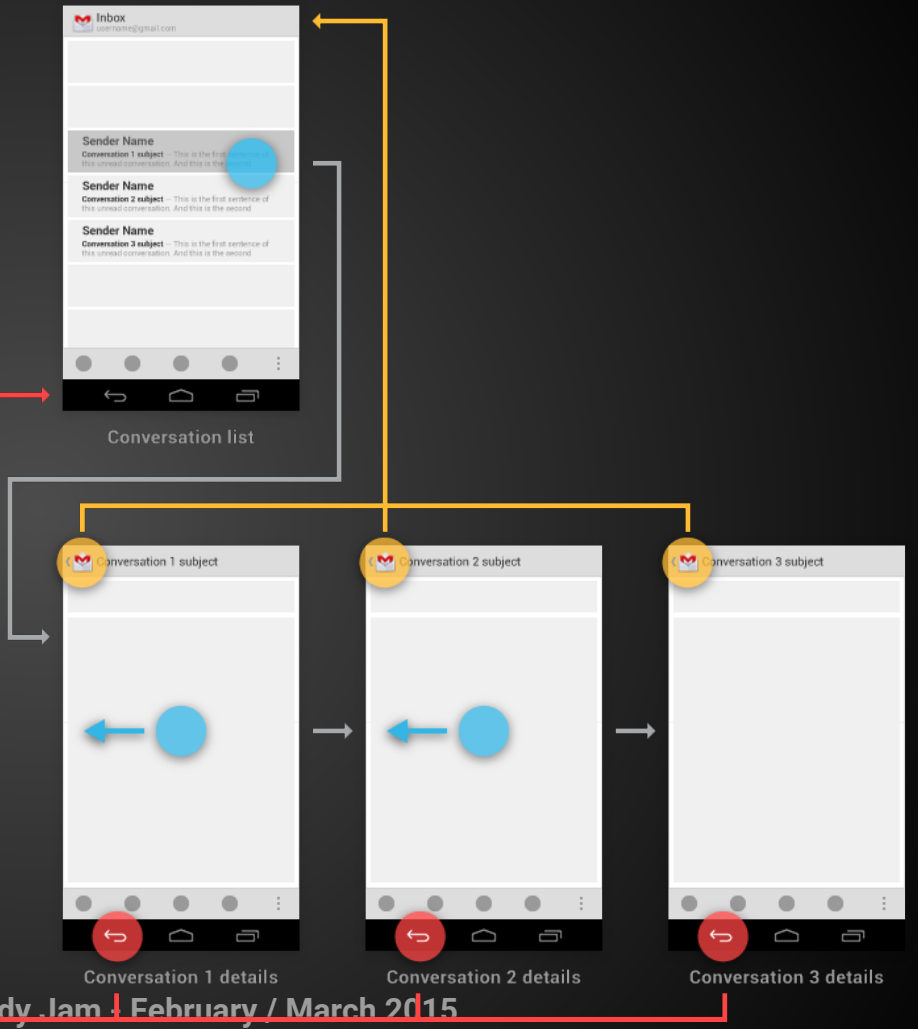
"withText"

"always"

"collapseActionView"



# Effective Navigation.
# Up vs Back



# Effective Navigation. Up vs Back

```
ActionBar actionBar = getActionBar();
actionBar.setDisplayHomeAsUpEnabled(true);

<!-- Parent activity support API level 16+ --> <android:
parentActivityName="com.example.myfirstapp.MainActivity" >

<!-- Parent activity meta-data to support API level 7+ -->
<meta-data
android:name="android.support.PARENT_ACTIVITY"
android:value="com.example.myfirstapp.MainActivity" />
```

# Effective Navigation. Antipatterns

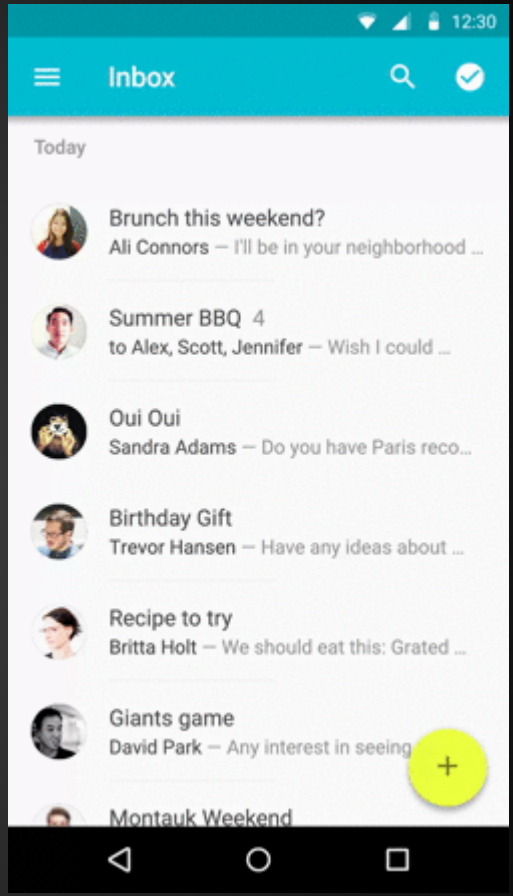
http://habrahabr.ru/post/213097/

# SwipeRefreshLayout

```
<android.support.v4.widget.SwipeRefreshLayout
    android:id="@+id/refresh"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <ListView
        android:id="@+id/list"
        android:layout_width="match_parent"
        android:layout_height="match_parent" />

</android.support.v4.widget.SwipeRefreshLayout>
```



# <include>

```
<include
        android:id=”@+id/news_title”
        android:layout_width=”match_parent”
        android:layout_height=”match_parent”
        layout=”@layout/title”/>
```

# <merge>

```
<merge xmlns:android="http://schemas.android.com/apk/res/android">

    <ImageView .../>
    <Button …/>
    <LinearLayout .../>

</merge>
```

# Loading Views On Demand

```
<ViewStub
    android:id="@+id/stub_import"
    android:inflatedId="@+id/panel_import"
    android:layout="@layout/progress_overlay"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"/>
```

# Loading Views On Demand

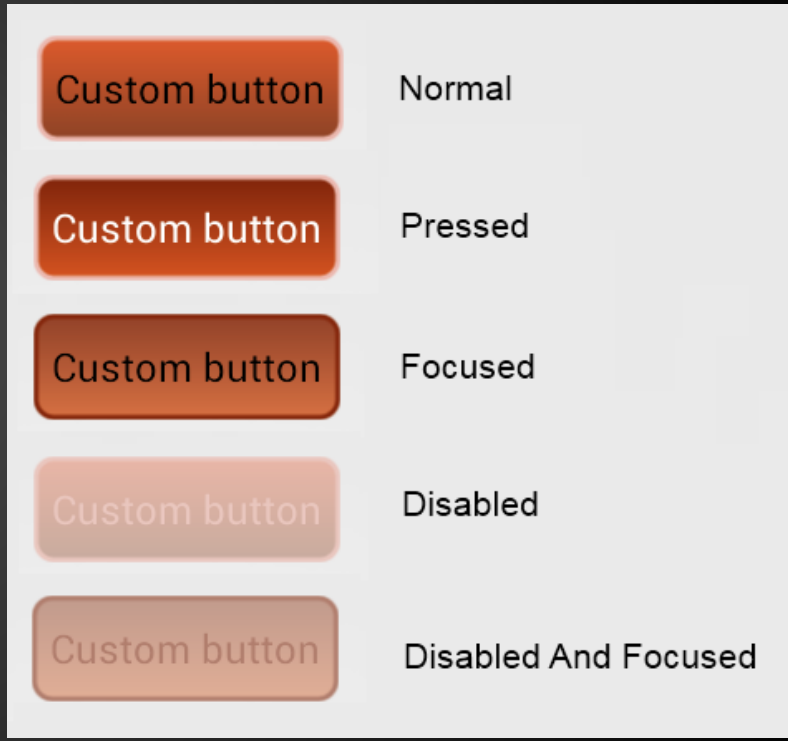
```
((ViewStub) findViewById(R.id.stub_import)).setVisibility(View.VISIBLE);
```

or

```
View importPanel = ((ViewStub) findViewById(R.id.stub_import)).inflate();
```

# Drawable.xml

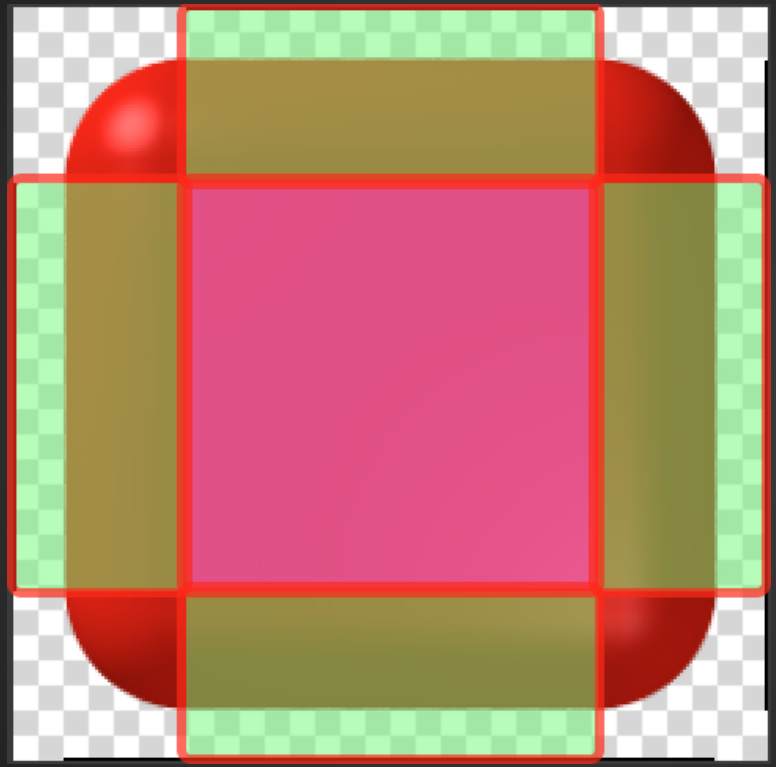
```
<selector >
    <item android: state_pressed="true"
android:drawable="@drawable/button_pressed" />
    <item android: state_focused="true"
android:drawable="@drawable/button_focused" />
    <item android: state_disabled="true"
android:drawable="@drawable/button_disabled" />
    <item android:drawable="
@drawable/button_normal" />
</selector>
```



# Draw 9-patch

# Draw 9-patch

# Draw 9-patch



# Draw 9-patch

# Home work

(по желанию) Пересохранить данные погоды, используя ORMLite

Написать тесты проверки работы обращений к БД

# Time for Q&A

Lesson 5