# 5. Activity

## Activity lifecycle

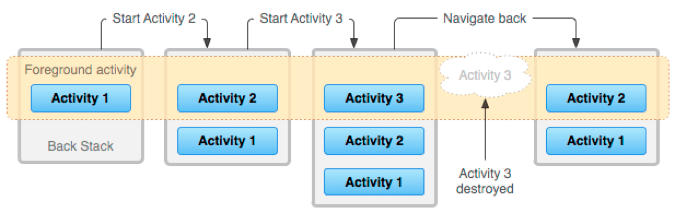
Activity - основная сущность приложения. По своей сути - это экран приложения.

* Может занимать весь экран или его часть.
* Может быть запущена из других компонент приложения или из другого приложения.
* Activity может возвращать результат



Полная версия жизненного цикла <https://github.com/xxv/android-lifecycle/blob/master/complete_android_fragment_lifecycle.png>

## Activity back stack



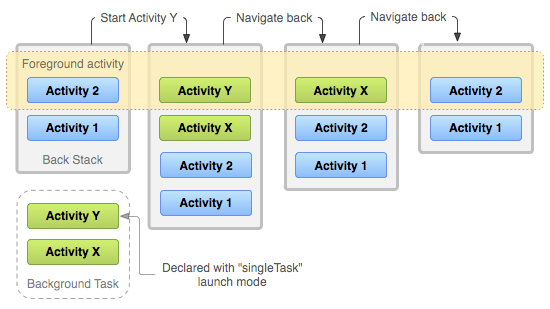
## Activity Launch Modes

**standart**(default mode) – при каждом запуске Activity создается новый экземпляр Activity и помещается на вершину back stack.

**singleTop** – если в момент запуска экземпляр Activity уже находится на вершине стека, то новый экземпляр не создается, вместо этого вызывается метод onNewIntent() у существующего экземпляра.

**singleTask** – Activity запускается в своем отдельном Task. Если экземпляр Activity уже существует, то у него вызывается метод onNewIntent(), а все Activity, лежащие в back stack поверх этого экземпляра – уничтожаются

**singleInstance** – то же, что и singleTask, но Activity является в своем таске единственной. Для приложений, которые состоят из одно активити. Используется крайне редко



## Пересоздание Activity

Android пересоздает Activity:

* При изменении конфигурации устройства, например когда
  + изменяется ориентация экрана
  + пользователь меняет язык системы в настройках Android

и т.п.

* При возврате пользователя к процессу, который был убит

Android для освобождения ресурсов.

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

public class MyActivity extends Activity {

public static final String KEY\_VISIBLE = "com.parallels.sample.key.VISIBLE";

private View mView;

@Override

public void onSaveInstanceState(Bundle outState) {

super.onSaveInstanceState(outState);

outState.putBoolean(KEY\_VISIBLE, mView.getVisibility() == VISIBLE);

}

@Override

public void onRestoreInstanceState(Bundle state) {

super.onRestoreInstanceState(state);

mView.setVisibility(

state.getBoolean(KEY\_VISIBLE) ? VISIBLE : GONE);

}

@Override

public void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

if (savedInstanceState != null) {

mView.setVisibility(

savedInstanceState.getBoolean(KEY\_VISIBLE) ? VISIBLE : GONE);

}

}

}

Parcelable - что это и зачем.

## Запуск активити для получения результата

public class MyActivity extends Activity {

...

static final int PICK\_CONTACT\_REQUEST = 0;

public boolean onKeyDown(int keyCode, KeyEvent event) {

if (keyCode == KeyEvent.KEYCODE\_DPAD\_CENTER) {

// When the user center presses, let them pick a contact.

startActivityForResult(

new Intent(Intent.ACTION\_PICK,

new Uri("content://contacts")),

PICK\_CONTACT\_REQUEST);

return true;

}

return false;

}

protected void onActivityResult(int requestCode, int resultCode,

Intent data) {

if (requestCode == PICK\_CONTACT\_REQUEST) {

if (resultCode == RESULT\_OK) {

// A contact was picked. Here we will just display it

// to the user.

startActivity(new Intent(Intent.ACTION\_VIEW, data));

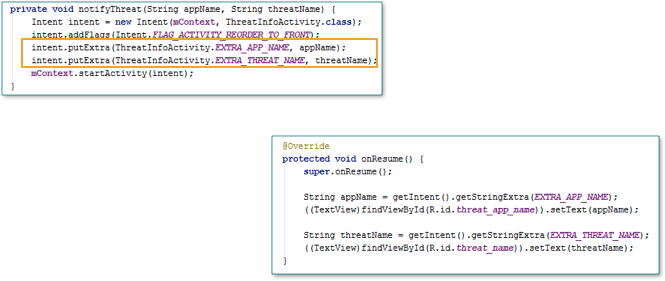
}

}

}

}

## Передача параметров в Активити



Подробнее для самостоятельного изучения:

<https://learndroid.e-legion.ru/lessons/4>

#### Задание:

Реализовать задания, которые записаны на диаграмме

<https://www.draw.io/#G1YcphHrAQkuh0IubGAzswwqtWWzZOLtJH>

<https://drive.google.com/file/d/1YcphHrAQkuh0IubGAzswwqtWWzZOLtJH/view?usp=sharing>

#### Полезные ссылки:

1. Официальная документация про Activity <https://developer.android.com/reference/android/app/Activity>
2. Полный жизненный цикл активити <https://github.com/xxv/android-lifecycle/blob/master/complete_android_fragment_lifecycle.png>
3. Пересоздание активити <https://developer.android.com/guide/topics/resources/runtime-changes.html>
4. Подробно про интенты и фильтры для самостоятельного просмотра: <https://learndroid.e-legion.ru/lessons/4>
5. Понятная статья про такс активити <https://habr.com/post/186434/>