Лекция №3.
по курсу «Мобильное программирование»

Москва 2020

КОМПОНЕНТЫ ПРИЛОЖЕНИЯ

- Application
- Activity
- Service
- BroadcastReceiver
- ContentProvider

АКТИВНОСТЬ

```
public class MainActivity extends AppCompatActivity {

  @Override
  protected void onCreate(Bundle savedInstanceState) {
     super.onCreate(savedInstanceState);
     setContentView(R.layout.activity_main);
  }
}
```

```
<activity android:name=".MainActivity">
  <intent-filter>
     <action android:name="android.intent.
                           action.MAIN"/>
     <category android:name="android.intent.
                           category.LAUNCHER"/>
  </intent-filter>
</activity>
```

SERVICE

```
public class CustomService extends IntentService {

  public CustomService() {
    super("CustomService");
  }

  @Override
  protected void onHandleIntent(Intent intent) {
    // TODO: load file
  }
}
```

```
<service
  android:name=".CustomService"
  android:exported="false"/>
```

BroadcastReceiver

```
public class CustomReceiver extends BroadcastReceiver {
  @Override
  public void onReceive(Context context, Intent intent) {
    // TODO: start service
  }
}
```

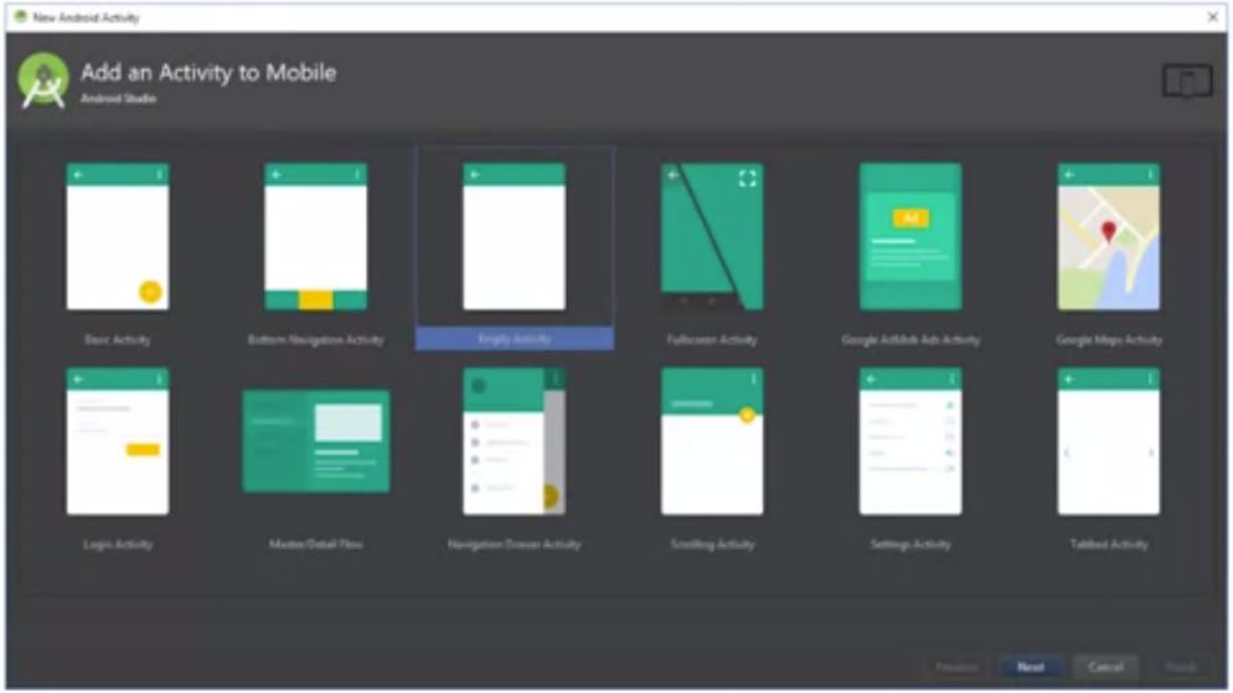
Доступ к системным возможностям устройства и ресурсам

```
AlarmManager am = (AlarmManager)
context.getSystemService(Context.
                                ALARM_SERVICE);
```

```
FingerprintManager fm= (FingerprintManager)
context.getSystemService(Context.
                                FINGERPRINT_SERVICE);
```

АКТИВНОСТЬ

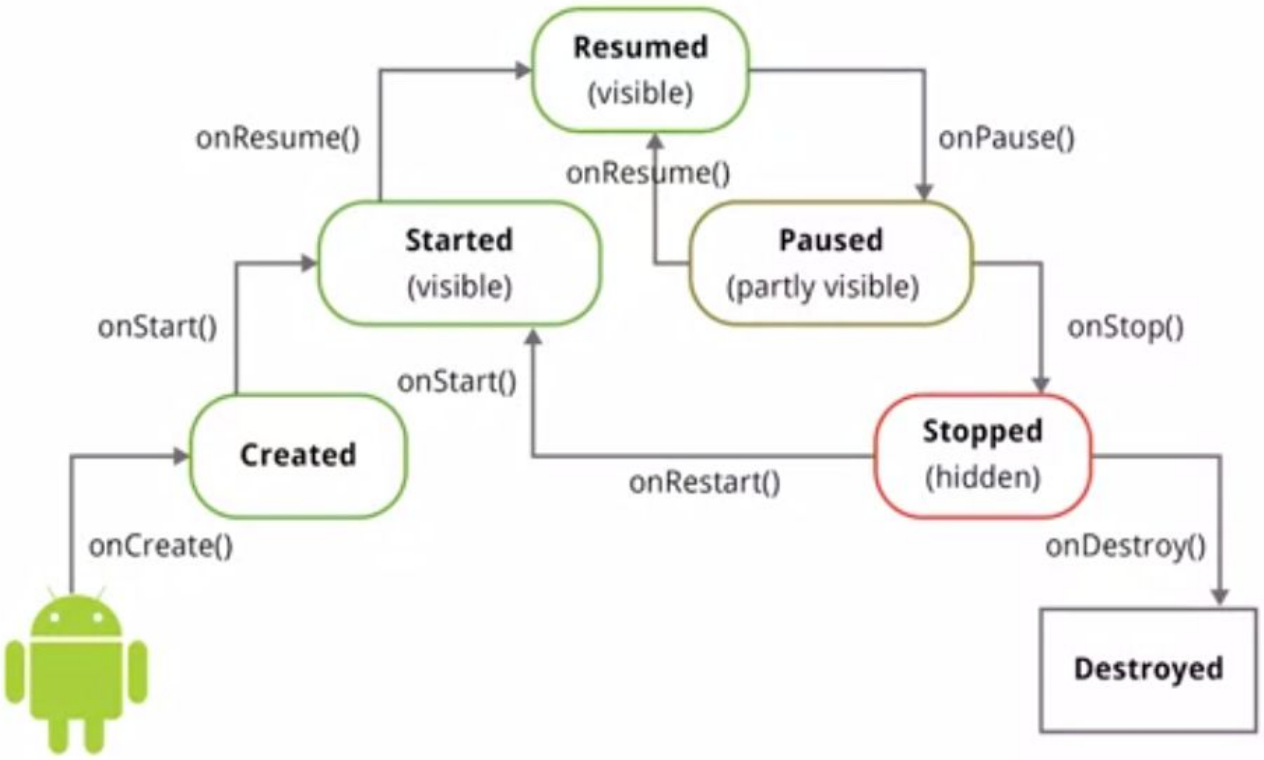
Главный компонент приложения, с помощью которой пользователь взаимодействует с приложением посредством UI — пользовательского интерфейса.



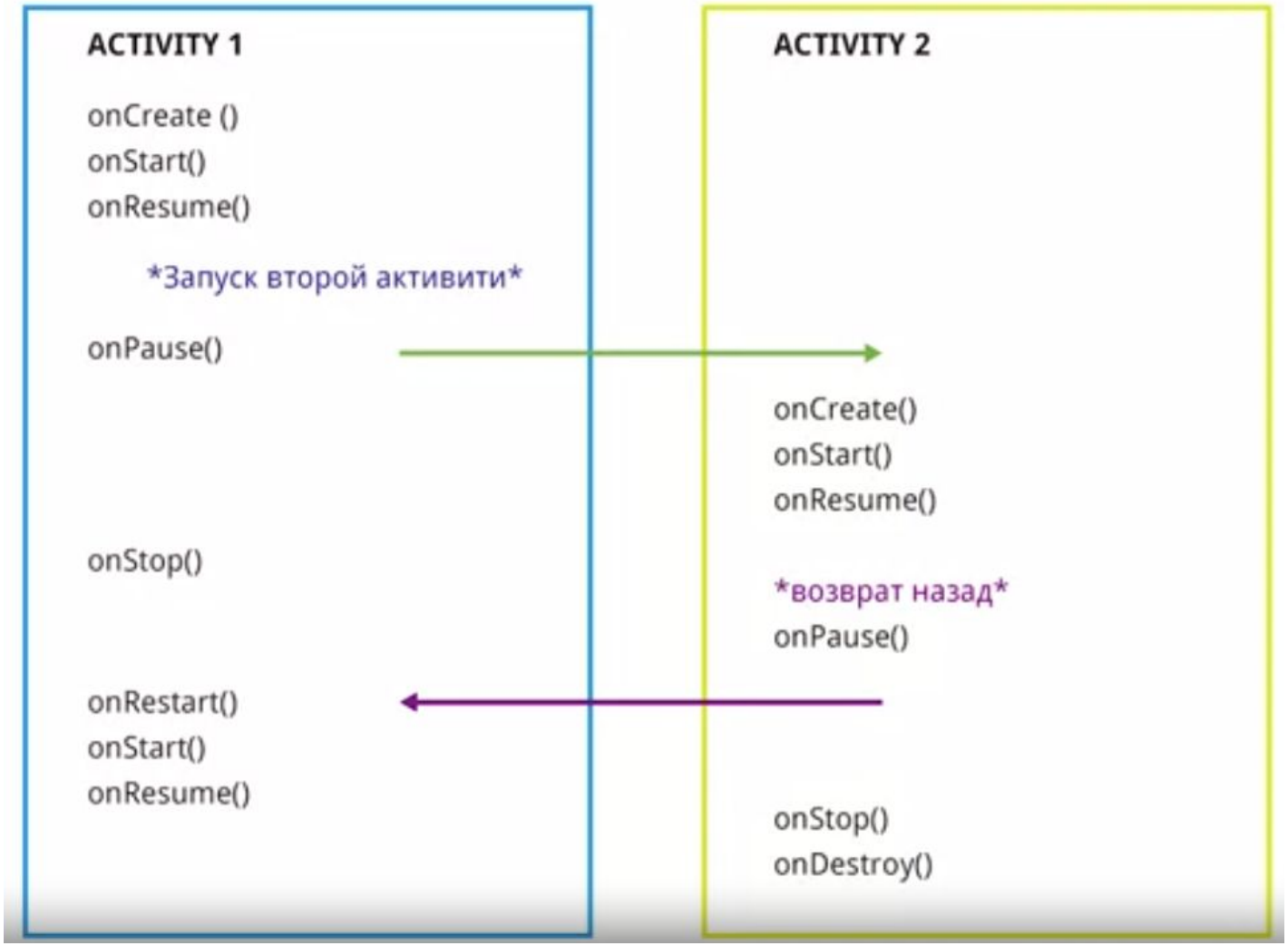
АКТИВНОСТЬ

```
<activity android:name=".MainActivity">
<intent-filter>
   <action android:name=
                "android.intent.action.MAIN" />

   <category android:name="android.intent.
                          category.LAUNCHER" />
</intent-filter>
</activity>
```

ЖИЗНЕННЫЙ ЦИКЛ АКТИВНОСТИ



СТЕК АКТИВНОСТЕЙ



Уничтожение активности

Система считает за нормальное поведение:

- нажатие кнопки “назад”
- вызов метода finish()

Система сохраняет стейт при следующих случаях:

- изменении конфигурации (поворот экрана, смена языка, доступность клавиатуры)
- уничтожение фоновой Activity при нехватке памяти для рабочей Activity

OnSaveInstanceState

Построение интерфейса Андроид

```
<View
  android:id="@+id/my_view"
  android:layout_width="wrap_content"
  android:layout_height="wrap_content"
            // другие атрибуты
/>
```

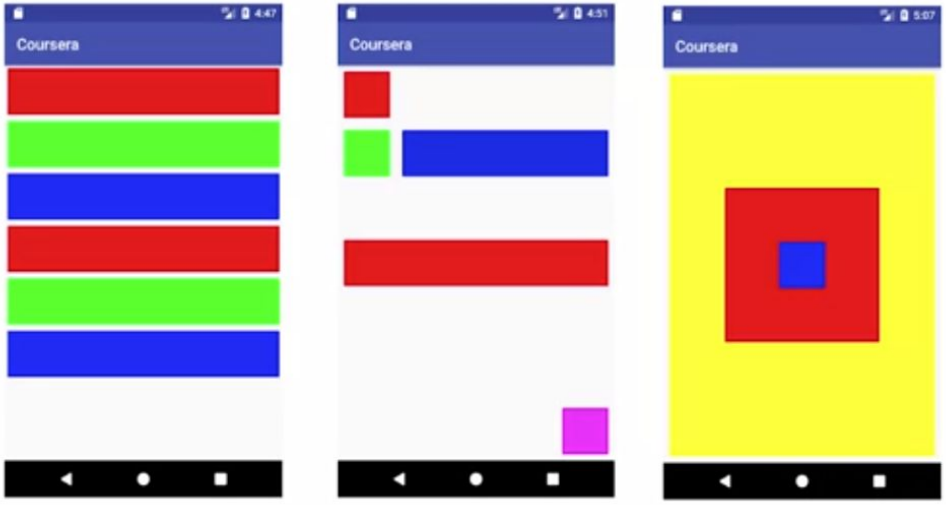
```
View myView = findViewById(R.id.my_view);
myView.setVisibility(View.VISIBLE);

View runtimeView = new View(context);
runtimeView.setVisibility(View.INVISIBLE);
```

ViewGroup, View

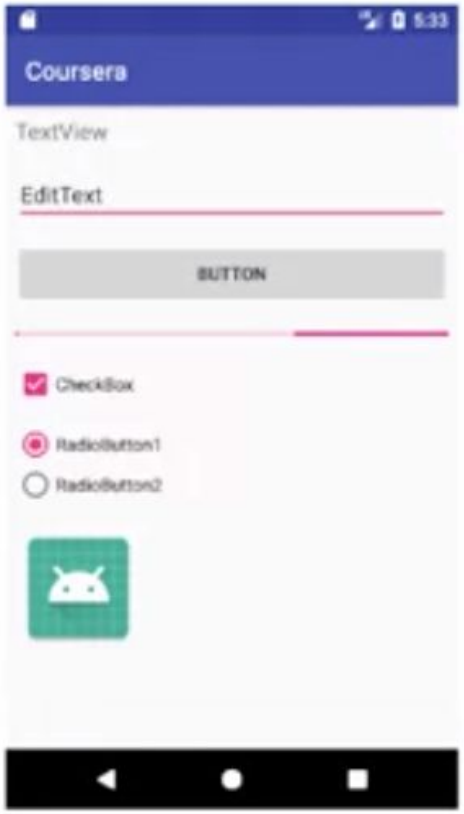
```
<ViewGroup>     //Иерархическая структура
                  разметки, XML правила
  <View/>
  <View/>
  <ViewGroup>
    <View/>
    <View/>
  </ViewGroup>
</ViewGroup>
```

ViewGroup, View

```
<ViewGroup>    //Иерархическая структура
                разметки, XML правила
  <View/>
  <View/>
  <ViewGroup>
    <View/>
    <View/>
  </ViewGroup>
</ViewGroup>
```



Стандартные компоненты



Элементы интерфейса из Android SDK.

Большая их часть настраивается и редактируется.

КОНТРОЛЬНАЯ РАБОТА

1. **Вариант 1**

   Напишите программу, которая находит максимальную последовательность последовательных равных элементов в массиве. Например: {1, 1, 2, 3, 2, 2, 2, 1} -> {2, 2, 2}

   Вариант №2

   Напишите программу, которая находит максимальную последовательность последовательно расположенных возрастающих целых чисел. Пример: {3, 2, 3, 4, 2, 2, 4} -> {2, 3, 4}.

2. **Вариант №1**

   Разработать класс для работы и представления рациональных чисел

3. **Вариант №2**

   Напишите программу для поиска последовательности соседних чисел в массиве, которая имеет сумму определенного числа S. Пример: {4, 3, 1, 4, 2, 5, 8}, S = 11 -> {4, 2, 5}.