

# Mobile Coding

PROF° RITIELLE SOUZA

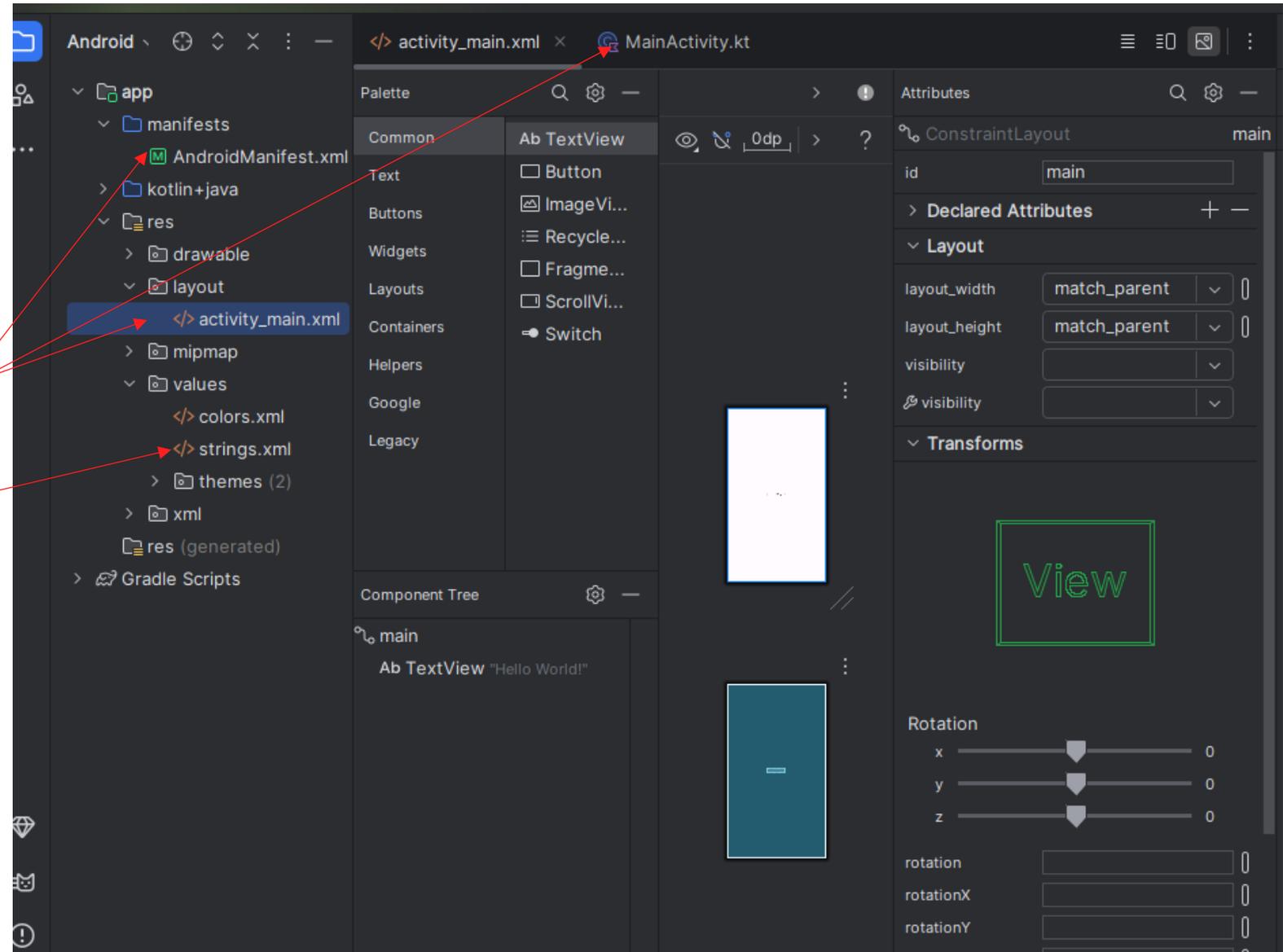




# ANDROID STUDIO

# ESTRUTURA DO PROJETO

Activity\_main.xml,  
MainActivity.kt,  
strings.xml,  
AndroidManifest.xml.



# ESTRUTURA DO PROJETO

AndroidManifest.xml é o “Sistema nervoso” do Android, é através dele que todas as permissões e configurações primordiais de nossa aplicação sejam executadas.

## **Diretório RES (resources)**

Dentro da estrutura de um projeto Android existe um diretório chamado res, onde dentro dele colocamos todos os recursos utilizados pelo programa android, imagens, sons, músicas e etc.

## **Diretório drawable**

Diretório utilizado com a finalidade de armazenar imagens utilizadas no aplicativo.

## **Diretório layout**

Armazena todos os arquivos referentes as telas de uma aplicação, normalmente em .xml.

## **Diretório Mipmap**

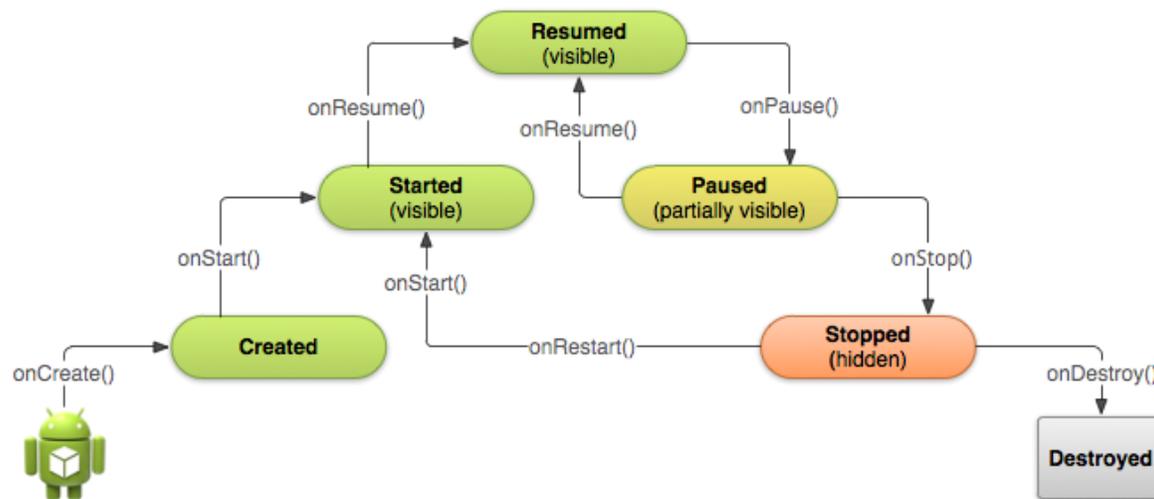
Diretório para imagens que serão ícones da aplicação.

# Activity

Activity ou atividade é algo definido que o usuário pode fazer. Estão associados as mais variadas funções. As atividades normalmente são associadas a uma única tela da aplicação.

## CICLO DE VIDA DA ACTIVITY (LIFECYCLE)

A activity é um componente de aplicação com um ciclo de vida específico. Sempre que a Activity muda de estado, o Android aciona um método correspondente.



# KOTLIN

- Friendly
- construído através da simplificação de soluções do mercado

```
class Person {  
    private var name: String  
    private var age: Int  
  
    constructor(name: String, age: Int) {  
        this.name = name  
        this.age = age  
    }  
  
    fun getName(): String {  
        return name  
    }  
  
    fun getAge(): Int {  
        return age  
    }  
}  
  
public class Person {  
    private final String name;  
    private final int age;  
  
    public Person(String name, int age) {  
        this.name = name;  
        this.age = age;  
    }  
  
    public String getName() {  
        return name;  
    }  
  
    public int getAge() {  
        return age;  
    }  
}
```

# KOTLIN

```
var a :Int = 2  
var b :String = "Hello"  
var c :Double = 2.0
```

# KOTLIN

```
var a :Int = 2  
var b :String = "Hello"  
var c :Double = 2.0
```

```
var a = 2  
var b = "Hello"  
var c = 2.0
```

Sem ";" no final das instruções

```
var bool : Boolean = funcaoQueRetornaBooleano()
```

# KOTLIN - classes

```
class Pessoa {  
    val nome: String  
  
    constructor(nome: String) {  
        this.nome = nome ← Boilerplate!  
    }  
  
    fun getNome(): String {  
        return nome ← Boilerplate!  
    }  
  
    fun tamanhoDoNome() {  
        println("Olá, meu nome tem ${nome.length} letras!")  
    }  
}
```

# KOTLIN – When exhaustivo

```
when (score) {  
    A,B -> println("Approved")  
    C,D -> println("Grey zone")  
    E -> println("Rejected")  
}
```

Else não necessário  
(todas possibilidades foram tratadas)

# Vamos criar um projeto

<https://replit.com/>