

# Autoria

Professor: Jackson

## Classes, atributos, métodos e activities (arquivo.java e arquivo.xml)

- Para a gente conseguir criar um aplicativo Android, precisamos criar arquivos .java e .xml para as classes, atributos, métodos e atividades.
- Os arquivos .java são responsáveis pela lógica do app, é onde as funções são implementadas.
- Os arquivos .xml são responsáveis pela interface do usuário. É onde os layouts são definidos e os elementos visuais são configurados.

## AndroidManifest.xml

- O AndroidManifest.xml é um arquivo que fornece informações essenciais sobre o aplicativo para o sistema operacional Android.
- Contém informações como o nome do pacote do app, as permissões necessárias, às atividades do aplicativo e outros elementos

## Método findViewById

- Esse método é usado para encontrar um elemento de interface do usuário pelo seu ID.
- É necessário chamar esse método após o método setContentView no arquivo .java.
- Exemplo:

```
Button myButton = findViewById(R.id.myButton);//acha na pasta "R" o botão com o id  
"myButton"
```

## Parâmetro View

- O parâmetro View é usado para lidar com eventos de interface do usuário, como cliques em botões.
- É passado como argumento para o método onClick no arquivo .java.
- Exemplo:

```
public void myButtonClick(View view) {  
  
    // Código a ser executado quando o botão é clicado  
  
}
```

## Toast

- O Toast é uma mensagem curta que aparece na tela para informar o usuário sobre uma ação realizada.
- É exibido por um curto período de tempo e desaparece automaticamente.
- Exemplo:

```
Toast.makeText(getApplicationContext(), "Hello World!", Toast.LENGTH_SHORT).show();
```

## OnCreate

- Ele é chamado quando a activity é criada, e é onde a maioria das inicializações deve ser feita.

### 3

- É um método importante no ciclo de vida de uma activity Android.
- Exemplo:

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

## Interpretação de código

- É importante saber como os arquivos .java e .xml se ligam e como os eventos de interface do usuário são tratados.
- É bom ter um conhecimento das classes, métodos e atributos disponíveis na plataforma Android.
- Para conseguir interpretar um código em Java no Android Studio, precisamos ter conhecimento da sintaxe da linguagem e da condição de um aplicativo Android.

