Android

Activity

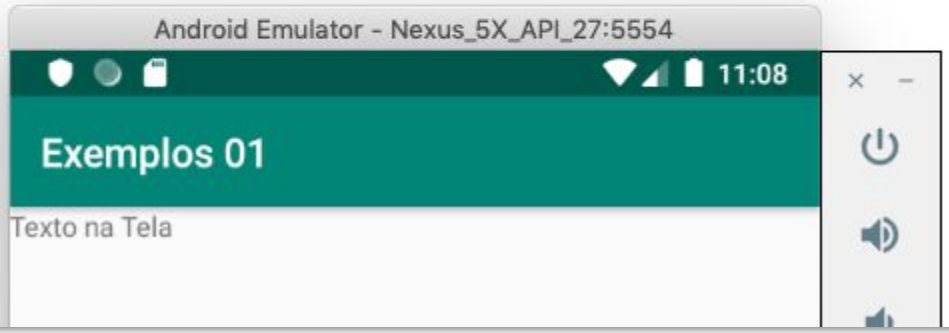
Professor Emerson Alencar
emerson@imd.ufrn.br

Activities

- Uma Activity representa uma tela da aplicação
- Representada por uma classe que herda de android.app.Activity

MainActivity.kt

```
1  package com.example.exemplos01
2
3  import ...
6
7  class MainActivity : AppCompatActivity() {
8
9      override fun onCreate(savedInstanceState: Bundle?) {
10         super.onCreate(savedInstanceState)
11
12
13         var view = TextView( context: this)
14
15         view.text = "Texto na Tela"
16
17         setContentView(view)
18
19     }
20 }
21
```



A classe Context

- Representa o **contexto de execução** da aplicação do Android
- Possui muitos métodos que são comumente chamados em aplicações
- Um objeto desta classe está normalmente disponível em vários lugares da aplicação
  - A classe **Activity** herda de **Context**
  - O Context é fornecido como parâmetro

A classe Context

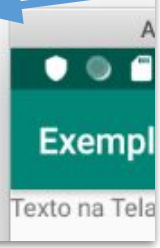
- O Android tem basicamente dois tipos de contextos:
  - **Contexto da activity:**
    - Cada activity da aplicação tem um contexto
  - **Contexto da aplicação:**
    - É único para a aplicação toda

- **Importante:** É preciso tomar cuidado na escolha do contexto para não causar memory leaks na aplicação (vazamento de memória)

A classe Context

MainActivity.kt

```
1  package com.example.exemplos01
2
3  import ...
6
7  class MainActivity : AppCompatActivity() {
8
9      override fun onCreate(savedInstanceState: Bundle?) {
10         super.onCreate(savedInstanceState)
11
12
13         var view = TextView( context: this)
14
15         view.text = "Texto na Tela"
16
17         setContentView(view)
18
19     }
20 }
21
```



Contexto da Activity: Essa viu sozinha não tem sentido sem Activity

```
3  import ...
7
8  class MainActivity : AppCompatActivity() {
9
10     override fun onCreate(savedInstanceState: Bundle?) {
11         super.onCreate(savedInstanceState)
12
13         val it = Intent(applicationContext,Main2Activity::class.java)
14
15
16     }
17 }
```

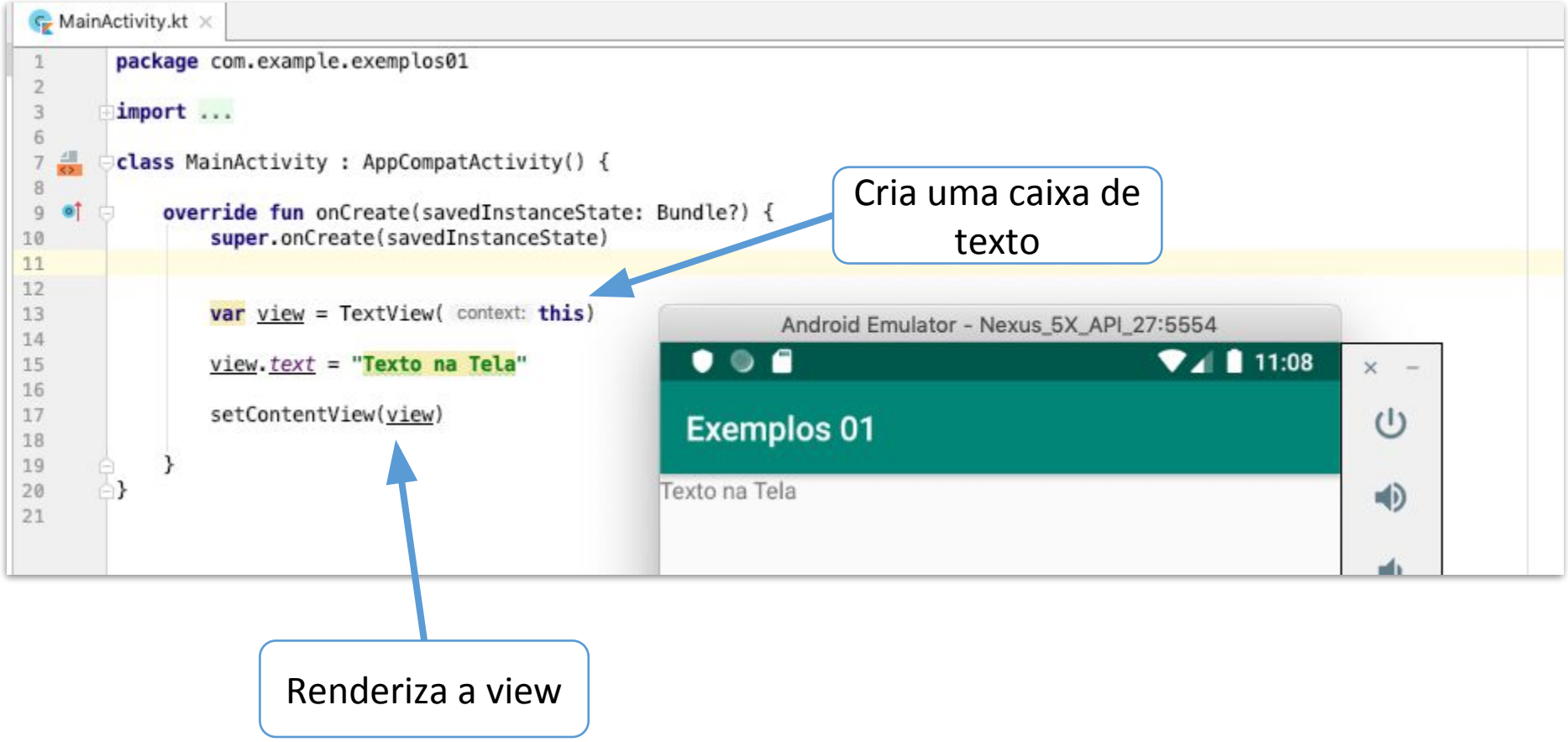
Contexto da Aplicação: Esse componente não tem relação com a Activity

Activities e Views

- **Views** são os componentes que se agrupam para montar a interface gráfica
- As **activities** e as views têm uma relação próxima
  - **Activities** representam a tela
  - **Views** representam o que é renderizado na tela

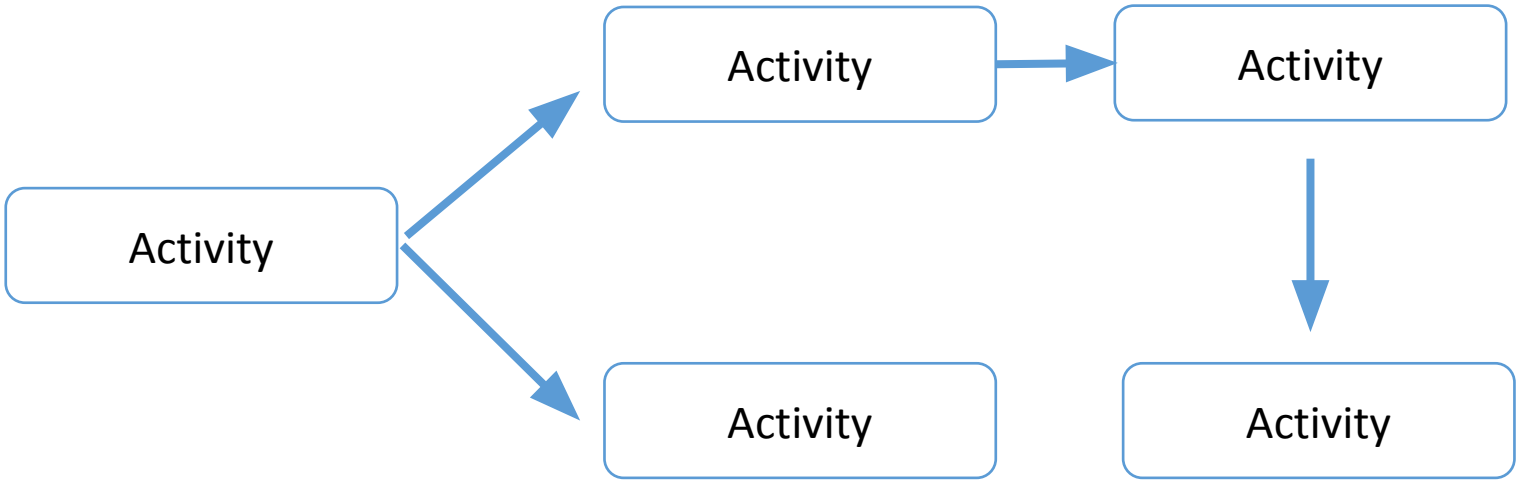
Activities e Views

- O método **setContentView()** da activity é utilizado para determinar qual view será renderizada



Invocando Activities

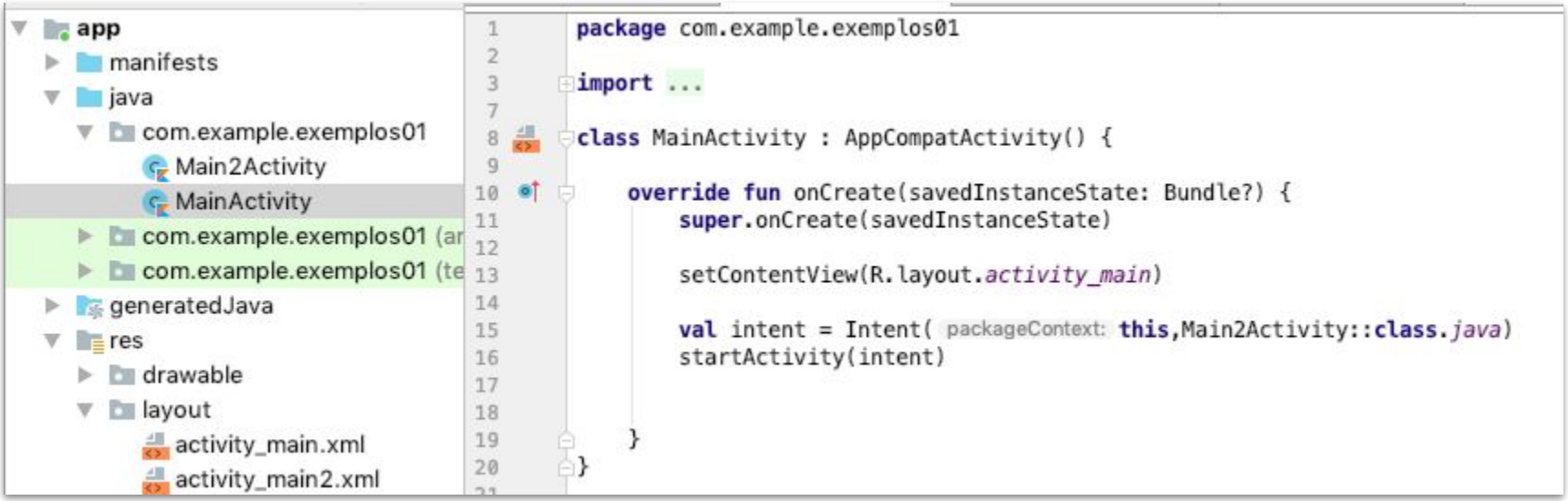
- Uma aplicação normalmente é composta por um conjunto de activities



A comunicação entre activities é feita através de uma intent

Invocando Activities

- Um objeto Intent é usada para disparar uma nova activity



Retornando Informações

- O método **startActivity()** chama outra activity
- Se esta nova activity for finalizada, a activity que fez a chamada não recebe nenhuma informação
- Para notificar a activity chamadora, é possível usar **startActivityForResult()**
- A nova activity carregada é considerada uma **sub-activity**, já que fica atrelada à activity que a chamou

Retornando Informações

```
8  class MainActivity : AppCompatActivity() {
9
10     override fun onCreate(savedInstanceState: Bundle?) {
11         super.onCreate(savedInstanceState)
12
13         setContentView(R.layout.activity_main)
14
15         val intent = Intent( packageContext: this,Main2Activity::class.java)
16         startActivityForResult(intent, requestCode: 1)
17
18     }
19
20     override fun onActivityResult(requestCode: Int, resultCode: Int, data: Intent?) {
21         super.onActivityResult(requestCode, resultCode, data)
22
23         //Aqui tratamos as informações de acordo com o requestCode
24     }
25 }
26
```

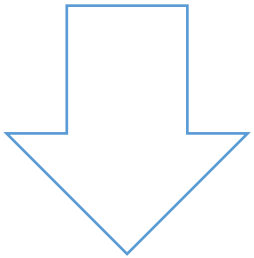
Informa um código de identificação

Sobrescreve o método onActivityResult, que é chamado quando a sub-activity é finalizada

Retornando Informações

```
setResult(10);
finish();
```

Activity invocada (sub-activity)
Define um código de resposta e finaliza a activty



onActivityResult(1,10,null)

Método na primeira activity

Passagem de Parâmetros

- É possível passar parâmetros de uma activity para outra usando um **Bundle (embrulho, pacote)**.

```
class MainActivity : AppCompatActivity() {

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)

        setContentView(R.layout.activity_main)

        val intent = Intent( packageContext: this,Main2Activity::class.java)

        intent.putExtra( name: "nome",  value: "Pedro")
        intent.putExtra( name: "idade",  value: 35)


        startActivity(intent)

    }
```

Passagem de parâmetro

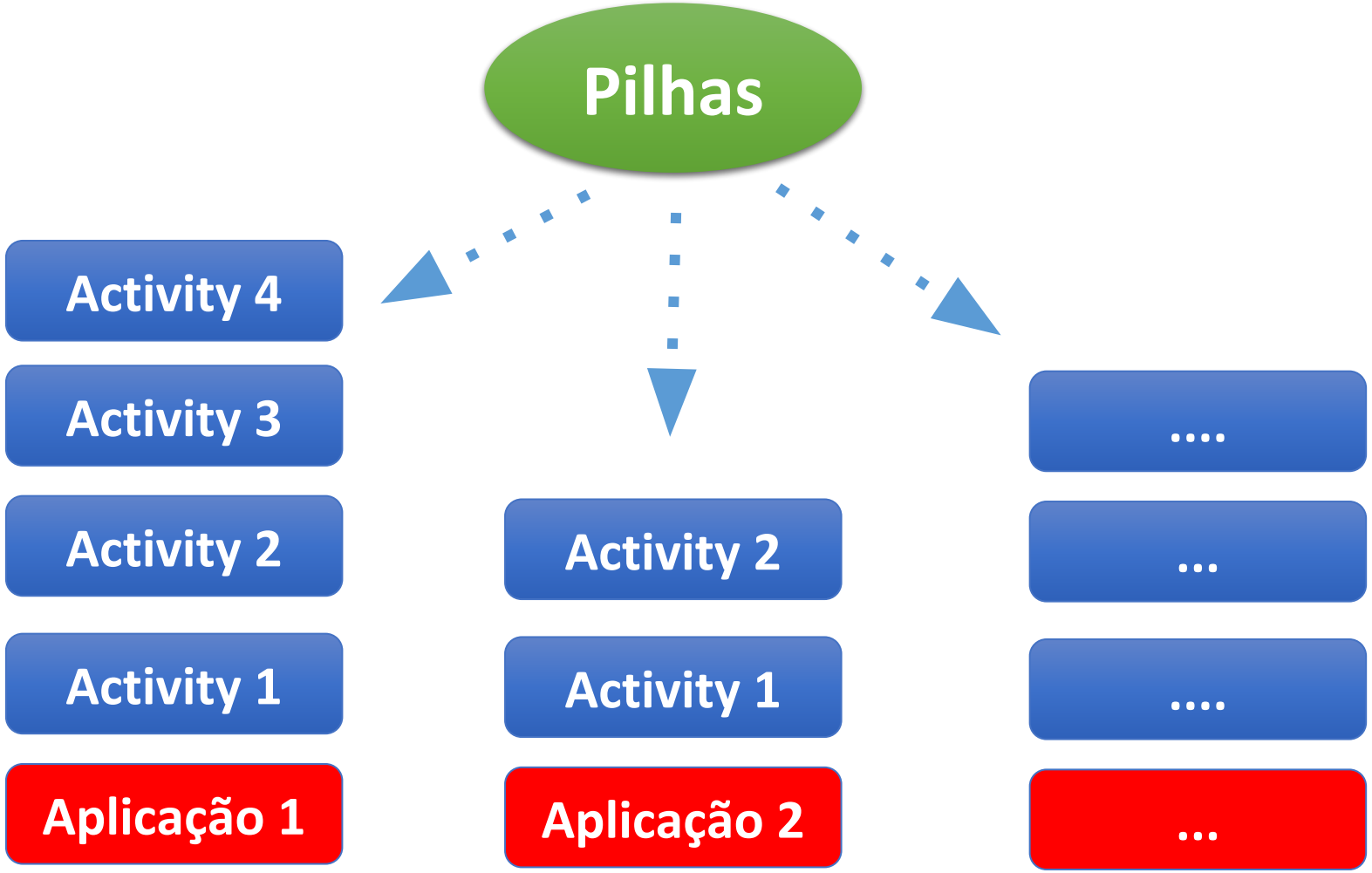
- A leitura do parâmetro é feita a partir da recuperação da intent que disparou a activity

```
class Main2Activity : AppCompatActivity() {

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)

        setContentView(R.layout.activity_main2)

        val nome = intent.getStringExtra( name: "nome")
        val idade = intent.getIntExtra( name: "idade", defaultValue: -1)


        Toast.makeText(applicationContext, text: "Nome: $nome e idade: $idade",Toast.LENGTH_LONG).show()



    }
}
```

As Activities na Memória

- As activities que fazem parte da sua aplicação são organizadas numa **pilha**
- A activity que está no **topo da pilha** é a activity mostrada na tela
- Toda vez que uma nova activity é invocada, ela passa a ficar no topo da pilha
- Quando o **botão voltar** é pressionado, a activity atual **é removida da pilha** e dá espaço para a activity que ocupa o topo da pilha ficar ativa

A Pilha de Activities

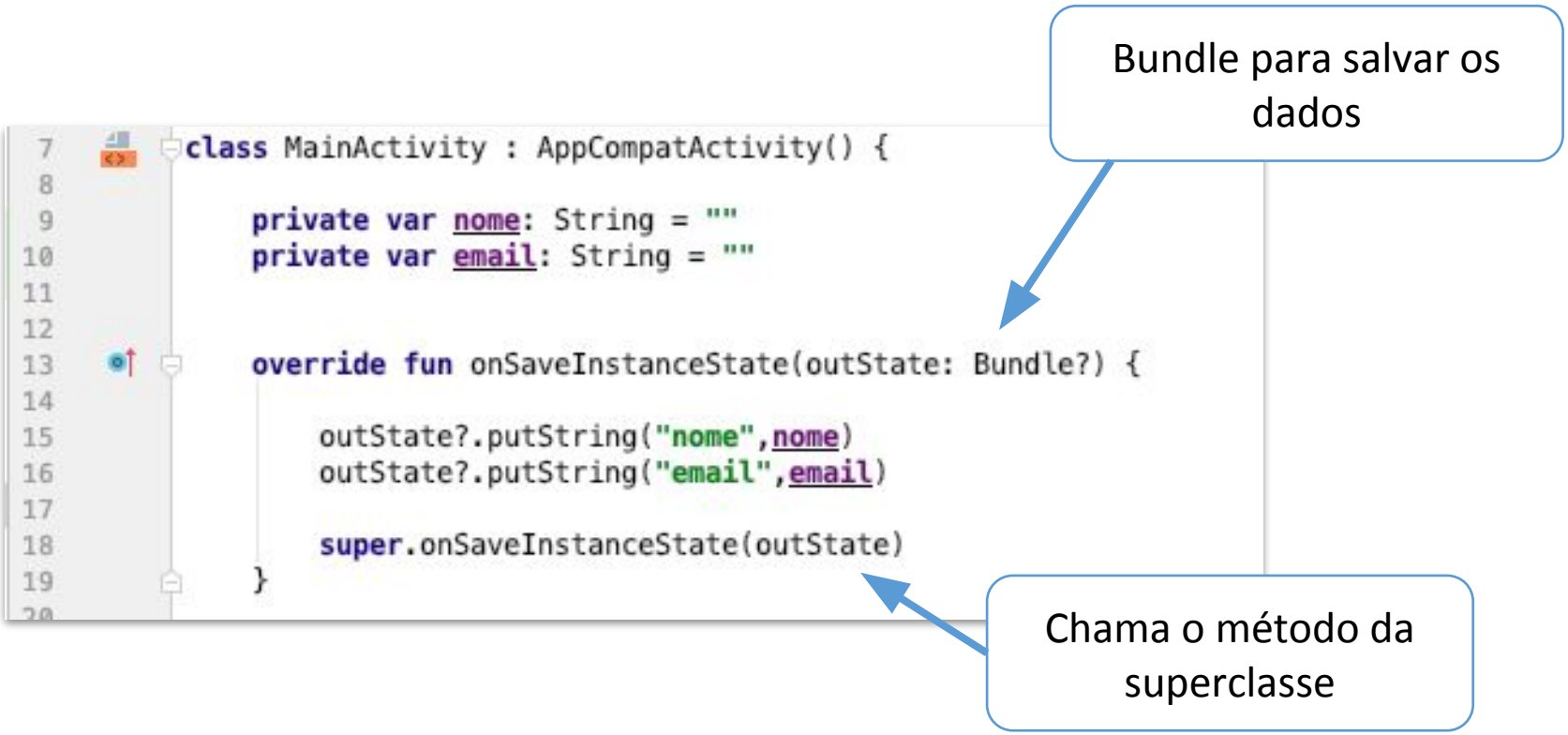


SALVANDO O ESTADO - ACTIVITY

- Uma activity pode ser destruída pelo Android para liberar recursos
- Quando ela for recriada, é possível que informações da activity sejam perdidas
- Para evitar que isto aconteça, é necessário salvar o estado da activity
- Não é preciso se preocupar com o estado das views, pois ele é gravado e depois recuperado
  - É preciso que a view tenha um ID especificado

O MÉTODO onSaveInstanceState()

- O Android chama o método onSaveInstanceState() na activity quando ela não fica mais visível.
- Quando a activity é destruída, o estado dela fica guardado.



RESTAURANDO O ESTADO-ACTIVTY

- Quando a activity for recriada, o Android chama o método onCreate() passando como parâmetro o bundle que contém os dados salvos.
- O método onRestoureInstanceState(Bundle) também é chamado (depois de onStart())

```
22    override fun onCreate(savedInstanceState: Bundle?) {
23        super.onCreate(savedInstanceState)
24
25        setContentView(R.layout.activity_main)
26
27        if (savedInstanceState != null){
28            this.nome = savedInstanceState.getString( key: "nome")
29            this.nome = savedInstanceState.getString( key: "email")
30        }
31
32        val edtNome = findViewById<EditText>(R.id.nome)
33        val edtEmail = findViewById<EditText>(R.id.email)
34
35    }
```

Se o Bundle não é **null**, a activity é restaurada

Obrigado!

Dúvidas?

Professor Emerson Alencar
emerson@imd.ufrn.br