



Desenvolvimento de Software para Dispositivos Móveis

Aula 18 - Flutter - Armazenamento



Professor: Anderson Almada

Exemplo prático

- Construir uma aplicação que utiliza de campos de texto
- Recupera as informações
- Salva no SharedPreferences
- Carrega do SharedPreferences

Crie o projeto

- Crie um projeto do zero e se utilize de um Stateful
- A propriedade **body** do Scaffold será um `SingleChildScrollView`

```
body: SingleChildScrollView (  
  child: Column (  
    children: <Widget>[  
  
    ],  
  )));
```

Primeiros elementos

- Vamos inserir primeiro os campos e texto.
- O primeiro e segundo widget da Column são Rows

Primeira linha

```
Row(  
  mainAxisAlignment: MainAxisAlignment.spaceEvenly,  
  children: <Widget>[  
    Flexible(  
      child: Text("Name:"),  
    ),  
    Flexible(  
      child: TextField(  
        controller: name),  
      )  
    ],  
),
```

Segunda linha

```
Row(  
  mainAxisAlignment: MainAxisAlignment.spaceEvenly,  
  children: <Widget>[  
    Flexible(  
      child: Text("Username:"),  
    ),  
    Flexible(  
      child: TextField(  
        controller: username),  
      )  
  ],  
)
```

Crie os controllers

- O próximo passo é criar os controllers que estão no TextField
- Crie os no começo da classe State

```
var name = TextEditingController ();  
var username = TextEditingController ();
```

Crie um botão

- Crie o botão, como filho de Column, que irá disparar a ação de adicionar

```
RaisedButton (  
  onPressed: () {  
    print(name.text + " " + username.text);  
  },  
  child: Text("Add"),  
)
```


Crie um model

- Crie um modelo User que irá representar as informações recuperadas
- Crie a pasta **models** em lib
- Crie o arquivo **user.dart**
- Acesse o site:
 - https://javiercbk.github.io/json_to_dart/

Crie um model

The screenshot shows a web browser window with the URL `javiercbk.github.io/json_to_dart/`. The page title is "JSON to Dart". Below the title, there is a instruction: "Paste your JSON in the textarea below, click convert and get your Dart classes for free." The interface is divided into two main sections. On the left, under the heading "JSON", there is a text area containing the following JSON:

```
{ 1 { 2 "username": "alnada", 3 "name": "Anderson" 4 }}
```

. Below the text area is a text input field containing the word "User". To the right of the input field is a blue button labeled "Generate Dart" and a checkbox labeled "Use private fields". Below these is a dark grey button labeled "Copy Dart code to clipboard". On the right side of the interface, the generated Dart code is displayed. It starts with a class definition:

```
class User {  String username;  String name;  User({this.username, this.name});  User.fromJson(Map<String, dynamic> json) {  username = json['username'];  name = json['name'];  }  Map<String, dynamic> toJson() {  final Map<String, dynamic> data = new Map<String, dynamic>();  data['username'] = this.username;  data['name'] = this.name;  return data;  } }
```

Crie um model

- O resultado você copia e cola no arquivo user.dart

Crie um list

- Crie uma lista para armazenar os dados inseridos pelos campos
- No começo da classe Stateful

```
var itens = new List<User>();
```

```
Construtor () {  
    itens = [];  
}
```

Crie o tile

```
ListTile _tile(String username, String name) => ListTile(  
  title: Row(  
    mainAxisAlignment: MainAxisAlignment.spaceEvenly,  
    children: <Widget>[  
      Flexible(  
        child: Text(username),  
      ),  
      Flexible(  
        child: Text(name),  
      ),  
    ],  
  ),  
);
```

Crie a lista

- Esse widget também será filho de Column

```
ListView.builder(  
  physics: const NeverScrollableScrollPhysics(),  
  shrinkWrap: true,  
  itemCount: widget.itens.length,  
  itemBuilder: (BuildContext ctxt, int index) {  
    return _tile(widget.itens[index].username,  
      widget.itens[index].name);  
  })
```

Altere a ação do botão

```
RaisedButton (  
  onPressed: () {  
    setState (() {  
      widget.itens.add(User(username: username.text, name:  
name.text));  
    });  
  
    print(name.text + " " + username.text);  
  },  
)
```

Usando o SharedPreferences

- Adicione a dependência
- Para isso, altere o arquivo pubspec.yaml

```
dependencies:
```

```
  flutter:
```

```
    sdk: flutter
```

```
shared_preferences: ^0.5.4+5
```


Usando o SharedPreferences

- Um método para carregar as informações

```
load() async {  
  var prefs = await SharedPreferences.getInstance();  
  var data = prefs.getString('data');  
  
  if (data != null) {  
    Iterable decoded = jsonDecode(data);  
    List<User> listItens = decoded.map((x) => User.fromJson(x)).toList();  
    setState(() {  
      widget.itens = listItens;  
    });  
  }  
}
```

Usando o SharedPreferences

- Um método para salvar as informações

```
save() async {  
    var prefs = await SharedPreferences.getInstance();  
    await prefs.setString('data', jsonEncode(widget.itens));  
}
```

Links importantes

- <https://flutter.dev/>



Dúvidas??

E-mail: almada@crateus.ufc.br