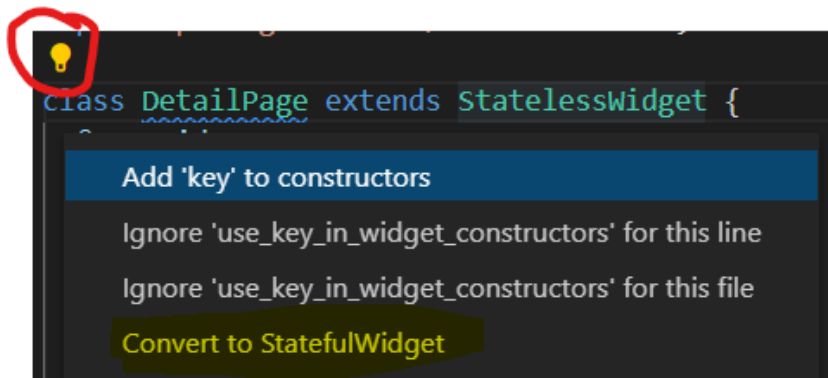


Login - php + mysql

Per tal de loguejar-nos a l'aplicació utilitzarem una mini api feta amb php i mysql.

Preparem la interfície

1. Si teniu la pàgina del login com a stateless widget, l'haurem de convertir a statefull ja que haurem d'interactuar amb el dispositiu.



2. Declararem els controladors dels dos camps de text dins l'State

```
class _LoginPageState extends State<LoginPage> {  
  var _user = TextEditingController();  
  _user.text="marta.millan";  
  var _pass = TextEditingController();  
}
```

3. Ara hem d'assignar aquests controladors als widgets dels camps de text `TextFormField`

```
TextFormField(  
  decoration: const InputDecoration(labelText: "Usuari"),  
  controller: _user,  
) ,
```

4. Farem que el botó, en fer click cridi a la funció login

```
onPressed: () {  
  login();  
}
```

```
},
```

ApiService.dart

A l'arxiu ApiService.ts crearem una funció que farà una crida a l'Api que hem creat

Farem un post a la url passant-li el username i el password inserits al formulari.

Si la resposta és 200, és a dir, la connexió ha anat bé analitzarem la resposta.

Si tenim un error ens mostrarà un missatge i isLogin es posarà en false.

Si ha anat bé mostrarà un missatge i isLogin es posarà a true.

El mètode retornarà un booleà

```
Future<bool> login(String user, String pass) async {  
  
    var url = "http://www.sundarabcn.com/flutter/login.php";  
  
    bool isLogin = false;  
  
    var response = await http.post(Uri.parse(url), headers: {'Accept':  
'application/json'}, body: {'username': user, 'password': pass});  
  
    if (response.statusCode == 200) {  
        var jsondata = json.decode(response.body);  
        if (jsondata["error"]) {  
            showToast(jsondata["message"]);  
            isLogin = false;  
        } else if (jsondata["success"]) {  
            showToast(jsondata["message"]);  
            isLogin = true;  
        }  
    } else {  
        showToast("Error de connexió");  
        isLogin = false;  
    }  
  
    return isLogin;  
}
```

```
}
```

Crearem una funció que es diu showToast que mostrarà un missatge a l'usuari

```
showToast(String message) {  
  Fluttertoast.showToast(  
    msg: message,  
    toastLength: Toast.LENGTH_LONG,  
    gravity: ToastGravity.CENTER,  
    timeInSecForIosWeb: 1,  
    backgroundColor: Colors.red,  
    textColor: Colors.white,  
    fontSize: 16.0);  
}
```

Per a fer-ho haurem d'importar la dependència al pubspec.yaml

```
fluttertoast:
```

i a l'arxiu .dart importarem la llibreria

```
import 'package:fluttertoast/fluttertoast.dart';
```

Exercici


1. Modifica el codi del botó del login per a que guardi en una variable el retorn de la funció creada anteriorment.
2. Si ha retornat un true fes que vagi a la següent pantalla.

Aquí teniu un exemple del que retorna l'Api si el login és correcte.

```
error:      false  
message:   "Login"  
success:   true  
username:  "marta.millan"  
id:        "15"
```

Shared preferences

A la pantalla del detall voldrem guardar com a preferida una imatge del dia. Aquesta és l'estructura de la BBDD:

#	Nom	Tipus	Col·lació
<input type="checkbox"/> 1	id 	int(11)	
<input type="checkbox"/> 2	idUser	int(11)	
<input type="checkbox"/> 3	date	varchar(128)	utf8_general_ci
<input type="checkbox"/> 4	explanation	text	utf8_general_ci
<input type="checkbox"/> 5	title	text	utf8_general_ci
<input type="checkbox"/> 6	url	text	utf8_general_ci

Com podeu observar aquest registre necessita de l'idUser i per tant necessitem guardar-lo després de fer el login. Ho farem mitjançant les Shared Preferences.

Passos

1. Importarem el mòdul de Shared Preferences al pubspec.yaml

```
shared_preferences
```

2. Només guardarem el uid si el login és correcte.
3. Haurem de declarar un objecte SharedPreferences i assignar-li un valor en format clau-valor

```
final prefs = await SharedPreferences.getInstance();  
await prefs.setString('id', jsondata["id"]);
```

4. A la pantalla on vulguem recuperar aquest valor farem una funció que ens retornarà aquest id. Ho hem de fer des d'una funció per a que aquesta sigui asíncrona.

```
getId() async {  
    final prefs = await SharedPreferences.getInstance();  
    final String? id = prefs.getString('id');  
}
```



Add Favorite

Aquesta és la taula per afegir informació a la base de dades. L'id és autoincrementable.

id	idUser	date	explanation	title	url	copyright
----	--------	------	-------------	-------	-----	-----------

URL:

```
var url = "http://www.sundarabcn.com/flutter/addData.php";
```

Mètode: POST

Read Favourite

URL:

```
var url = "http://sundarabcn.com/flutter/readData.php?idUser=_____";
```

Mètode: GET