FLUTTER

GETX

GETX

An extra-light and powerful solution for Flutter. It combines high-performance state management, intelligent dependency injection, and route management quickly and practically

PERFORMANCE

GetX is focused on performance and minimum consumption of resources. GetX does not use Streams or ChangeNotifier

PRODUCTIVITY

GetX uses an easy and pleasant syntax. No matter what you want to do, there is always an easier way with GetX

ORGANIZATION

GetX allows the total decoupling of the View, presentation logic, business logic, dependency injection, and navigation.

GetX
Snackbar

```dart
1  import 'package:flutter/material.dart';
2  import 'package:get/get.dart';
3  class HomePage extends StatelessWidget {
4    const HomePage({super.key});
5    @override
6    Widget build(BuildContext context) {
7      return Scaffold(
8        appBar: AppBar(title: Text('Counter App')),
9        body: Center(
10         child: ElevatedButton(onPressed: (){
11           Get.snackbar(
12             "Hi","I am Rabbil Hasan",
13             snackPosition:SnackPosition.BOTTOM,
14             showProgressIndicator: true ,
15             colorText: Colors.white,
16             backgroundColor: Colors.green,
17             borderRadius: 2,
18             margin: EdgeInsets.all(20),
19             padding: EdgeInsets.all(10),
20             icon:Icon(Icons.access_time,color: Colors.white),
21             duration: Duration(seconds: 10),
22             isDismissible: true,
23             animationDuration: Duration(milliseconds: 400)
24           );
25         },child: Text("Getx Snack Bar")),
26       )
27     );
28   }
29 }
30
```

GetX Dialog

```dart
        Get.defaultDialog(

          title: "Delete",
          titlePadding: EdgeInsets.all(2),
          titleStyle: TextStyle(color: Colors.white),

          middleText: "Onece delete, you can't get it back",
          middleTextStyle: TextStyle(color: Colors.white),

          backgroundColor: Colors.green,
          radius: 10,

          textConfirm: "Confirm",
          textCancel: "Cancel",
          cancelTextColor: Colors.white,
          confirmTextColor: Colors.white,
          buttonColor: Colors.red,
          barrierDismissible: false,

          onCancel: ()=>{Get.back(closeOverlays: true)},
          onConfirm: ()=>{Get.back(closeOverlays: true)},

          content: Column(
            children: [
              Container(child:Text("Hello 1")),
              Container(child:Text("Hello 2")),
              Container(child:Text("Hello 3")),
            ],
          )
        );
```

GETX
STORAGE

```dart
16              ElevatedButton(onPressed: (){
17                GetStorage().write("Name", "Rabbil");
18              },child: Text("Getx write name")),
19
20              ElevatedButton(onPressed: (){
21                var Name=GetStorage().read("Name");
22                Get.snackbar("Name",Name);
23              },child: Text("Getx read name")),
24
25              ElevatedButton(onPressed: (){
26                GetStorage().erase();
27              },child: Text("Getx Clear")),
```

HomePage.dart

GETX ROUTING

```
body:Center(
  child:  ElevatedButton(onPressed: (){
    // Get.to(SecondPage());
    // Get.off(SecondPage());
    // Get.offAll(SecondPage());
  },child: Text("Go Second Page"))
)
```

01 Navigate to a new screen:
Get.to(NextScreen());

02 Navigate to new screen with name.
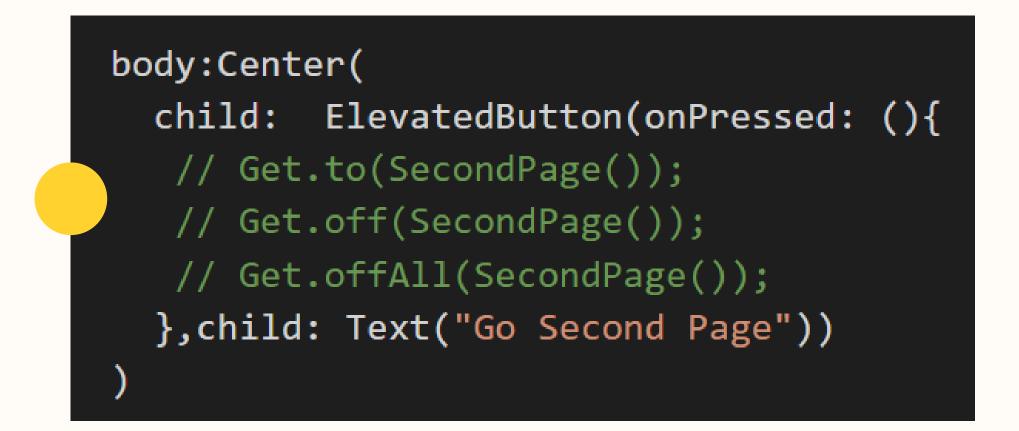Get.toNamed('/details');

03 To close snackbars, dialogs, bottomsheets, or anything
Get.back();

04 To go to the next screen and no option to go back to the previous screen
Get.off(NextScreen());

05 To go to the next screen and cancel all previous routes
Get.offAll(NextScreen());

GetX Named Route

```dart
14    return GetMaterialApp(
15      initialRoute: '/',
16      getPages: [
17        GetPage(name: '/', page: () => FirstPage()),
18        GetPage(
19            name: '/second',
20            page: () => SecondPage(),
21            transition: Transition.leftToRight
22        )
23      ],
24    );
```

```dart
body:Center(
  child:  ElevatedButton(onPressed: (){
    // Get.to(SecondPage());
    // Get.off(SecondPage());
    // Get.offAll(SecondPage());
    Get.toNamed('/second');
  },child: Text("Go Second Page"))
```

GetX
Bottom Sheet

```dart
OpenMyBottomSheet(){
    Get.bottomSheet(
        Container(
            height: 150,
            color: Colors.green,
            child:Column(
                children: [
                    Text('R'),
                    Text('A'),
                    Text('B'),
                    Text('B'),
                    Text('I'),
                    Text('L'),
                ],
            )
        ),
        barrierColor: Colors.red[10],
        isDismissible: true,
        enableDrag: true,
    );
}
```

CounterPage.dart

GetX Counter App

```dart
// CounterPage.dart
9    Widget build(BuildContext context) {
10     return Scaffold(
11       appBar: AppBar(title: Text('Counter App')),
12       body: Center(
13         child: Obx(()=>
14           Text("Clicks: ${c.count}")
15         )
16       ),
17       floatingActionButton: FloatingActionButton(
18         onPressed: (){
19           c.increment();
20         },
21         child: Icon(Icons.add),
22       ),
23     );
24   }
```

```dart
// CounterState.dart
3  class CounterController extends GetxController{
4    var count = 0.obs;
5    increment(){
6      count=count+1;
7    }
8  }
9
```

NOW WE ARE GOING TO CREATE

CRUD APPLICATION
USING REST API