



Flutter ক্লাস নোট - (Week 2)

আজকের ক্লাসে আমরা কিছু গুরুত্বপূর্ণ Flutter কনসেপ্ট নিয়ে আলোচনা করেছি। আমি যেভাবে তোমাদের বুঝিয়েছি, সেইভাবে নিচে সাজিয়ে দিলাম:



Navigator.push এবং Navigator.pop

◆ Navigator.push(context, route)

আমরা যখন একটি নতুন পেজে যেতে চাই, তখন `Navigator.push` ব্যবহার করি।

✦ উদাহরণ:

```
Navigator.push(  
  context,  
  MaterialPageRoute(builder: (context) => SecondPage()),  
);
```

◆ Navigator.pop(context)

আমরা যখন আগের পেজে ফিরে যেতে চাই, তখন `Navigator.pop` ব্যবহার করি।

✦ উদাহরণ:

```
Navigator.pop(context);
```



GestureDetector

◆ GestureDetector দিয়ে আমরা যেকোনো Widget এর উপর Tap, Double Tap, Long Press ইত্যাদি Action নিতে পারি।

✦ উদাহরণ:

```
GestureDetector(  
  onTap: () {  
    print("Tapped!");  
  },  
  child: Container(  
    color: Colors.blue,  
    height: 100,  
    width: 100,  
    child: Center(child: Text("Tap Me")),  
  ),  
);
```

```
),  
)
```

👶 child ও children কী?

- ◆ `child` হচ্ছে একক Widget।
- ◆ `children` হচ্ছে একাধিক Widget এর লিস্ট।

✦ উদাহরণ:

```
// একটিমাত্র Widget  
Container(  
  child: Text("Hello"),  
)  
  
// একাধিক Widget  
Column(  
  children: [  
    Text("One"),  
    Text("Two"),  
  ],  
)
```

↔ Row, Column

- ◆ `Row` : Widgets গুলোকে Horizontally (ডান-বাম) সাজায়।
- ◆ `Column` : Widgets গুলোকে Vertically (উপর-নিচ) সাজায়।

✦ উদাহরণ:

```
Row(  
  children: [  
    Icon(Icons.home),  
    Text("Home"),  
  ],  
)  
  
Column(  
  children: [  
    Text("Name"),  
    Text("Email"),  
  ],  
)
```

SingleChildScrollView

- ◆ যখন Content অনেক বড় হয় এবং স্ক্রল করতে হয়, তখন `SingleChildScrollView` ব্যবহার করি।

✦ উদাহরণ:

```
SingleChildScrollView(  
  scrollDirection: Axis.vertical, // অথবা Axis.horizontal  
  child: Column(  
    children: [  
      Text("Line 1"),  
      Text("Line 2"),  
      // আরও অনেক কিছু...  
    ],  
  ),  
)
```

Stateful Widget দিয়ে Increment-Decrement

- ◆ Stateful Widget দিয়ে আমরা UI তে পরিবর্তন আনার জন্য state change করতে পারি।
- ◆ আমরা Counter তৈরি করে দেখিয়েছি কিভাবে `+` ও `-` বাটন কাজ করে।

✦ উদাহরণ:

```
int count = 0;  
  
ElevatedButton(  
  onPressed: () {  
    setState(() {  
      count++;  
    });  
  },  
  child: Text("Increment"),  
)  
  
ElevatedButton(  
  onPressed: () {  
    setState(() {  
      count--;  
    });  
  },  
  child: Text("Decrement"),  
)  
  
Text("Count: $count")
```

Homework: একটি Full Functional Calculator App তৈরি করো

লক্ষ্য:

তোমাদের কাজ হলো এমন একটি Calculator অ্যাপ বানানো যেটা নিচের সব অপারেশন করতে পারে:

- ◆ যোগ (+)
- ◆ বিয়োগ (−)
- ◆ গুণ (×)
- ◆ ভাগ (÷)
- ◆ শতকরা (%)
- ◆ Clear (C)
- ◆ Equals (=) ফলাফল দেখানোর জন্য

যেসব Widget দরকার হবে:

- ◆ `TextEditingController`

ইউজার ইনপুট Field এর মান ধরার জন্য।

উদাহরণ:

```
TextEditingController inputController = TextEditingController();
```

- ◆ `TextField` (অথবা শুধু Text Widget)

ইউজার এর ইনপুট বা ফলাফল দেখানোর জন্য।

উদাহরণ (শুধু ফলাফল দেখাতে চাইলে):

```
Text("Result: $displayText", style: TextStyle(fontSize: 32))
```

- ◆ `GridView`, `Row`, `Column`

ক্যালকুলেটরের বাটনগুলো সাজানোর জন্য।

উদাহরণ:

```
GridView.count(  
  crossAxisCount: 4,  
  children: [  
    buildButton("C"),  
    buildButton("0"),  
    buildButton("%"),
```

```

        buildButton("/"),
        // বাকি বাটনগুলো...
    ],
)

```

◆ Custom Function `buildButton(String text)`

সব বাটন তৈরি করার জন্য একটি ফাংশন বানাও যেটা GestureDetector দিয়ে থাকবে:

✦ উদাহরণ:

```

Widget buildButton(String text) {
    return GestureDetector(
        onTap: () {
            onPressed(text);
        },
        child: Container(
            margin: EdgeInsets.all(8),
            decoration: BoxDecoration(
                color: Colors.grey[800],
                borderRadius: BorderRadius.circular(10),
            ),
            child: Center(
                child: Text(
                    text,
                    style: TextStyle(fontSize: 28, color: Colors.white),
                ),
            ),
        ),
    );
}

```

◆ `onButtonPressed(String buttonText)` Function

✦ উদাহরণ:

```

void onButtonPressed(String text) {
    setState(() {
        if (text == "C") {
            input = "";
            result = "";
        } else if (text == "=") {
            // এখানে যোগ, বিয়োগ, গুণ, ভাগ হিসাব করে result বের করো
        } else {
            input += text;
        }
    });
}

```

Hint:

- তুমি চাইলে `math_expressions` প্যাকেজ ব্যবহার করে ইনপুট থেকে সরাসরি Result বের করতে পারো।
 - অথবা তুমি নিজেই String পার্স করে অপারেশন করতে পারো।
-

Extra Tips:

- Calculator App টি Responsive করার চেষ্টা করো
 - "Backspace" অপশন যোগ করতে পারো
 - একটি Light/Dark Mode toggle করতে পারো
-

Note : পরবর্তী ক্লাসে তোমাদের বানানো Calculator App দেখে বোঝাবো কে কতটা বুঝেছে। ❤️ চেষ্টা করো প্রফেশনাল Calculator এর মতো বানাতে (যেমনটা আমি ছবিতে দেখিয়েছি)।

<https://github.com/user-attachments/assets/8c292f4f-ee0a-493a-a785-69516073352a>