

UAS
MOBILE PROGRAMMING
Ade Putra Prima Suhendri, S.Kom, M.Kom

SOAL

1. Jelaskan apa yang dimaksud dengan Mobile Programming? Point 5
2. Jelaskan apa yang dimaksud dengan User Interface (UI)? Point 5
3. Jelaskan apa yang dimaksud dengan API? jelaskan fungsinya! Point 5
4. Jelaskan perbedaan Native dan Hybrid pada mobile programming? Point 5
6. Jelaskan apa fungsi github! Point 5
7. Apa output dari script berikut ! Point 10:

```
ListView.builder(  
  itemCount: 10,  
  itemBuilder: (context, i){  
    return Text("$i");  
  },  
);
```

8. Apa output dari script berikut ! Point 10:

```
int timesTwo(int x) {  
  return x * 2;  
}  
  
int timesFour(int x) => timesTwo(timesTwo(x));  
  
int runTwice(int x, int Function(int) f) {  
  for (var i = 0; i < 2; i++) {  
    x = f(x);  
  }  
  return x;  
}  
  
void main() {  
  print("4 times two is ${timesTwo(4)}");  
  print("4 times four is ${timesFour(4)}");  
  print("2 x 2 x 2 is ${runTwice(2, timesTwo)}");  
}
```

9. Tuliskan sintak cara parsing JSON pada flutter ! Poin 55

Nama : Nur Syifa Fauziah
Nim : 181011401055
Kelas : 06TPLE011
Mata Kuliah : Mobile Programming

1. Mobile programming adalah proses pembuatan aplikasi untuk perangkat mobile baik aplikasi yang bersifat offline maupun online. **Android** merupakan sistem operasi berbasis linux yang bahasa pemrograman aplikasinya dapat kita buat menggunakan java. Seperti namanya **Mobile Programing** adalah pemrograman yang diterapkan di **mobile**/Handphone atau Laptop.

2. User Interface adalah tampilan visual sebuah produk yang menjembatani sistem dengan pengguna (user). Tampilan UI dapat berupa bentuk, warna, dan tulisan yang didesain semenarik mungkin. Secara sederhana, UI adalah bagaimana tampilan sebuah produk dilihat oleh pengguna

3. API adalah singkatan dari Application Programming Interface yaitu sebuah software yang memungkinkan para developer untuk mengintegrasikan dan mengizinkan dua aplikasi yang berbeda secara bersamaan untuk saling terhubung satu sama lain.

Fungsi : penggunaan dari API adalah untuk saling berbagi data antar aplikasi yang berbeda tersebut,

Tujuan penggunaan API lainnya yaitu untuk mempercepat proses pengembangan aplikasi dengan cara menyediakan sebuah function yang terpisah sehingga para developer tidak perlu lagi membuat fitur yang serupa.

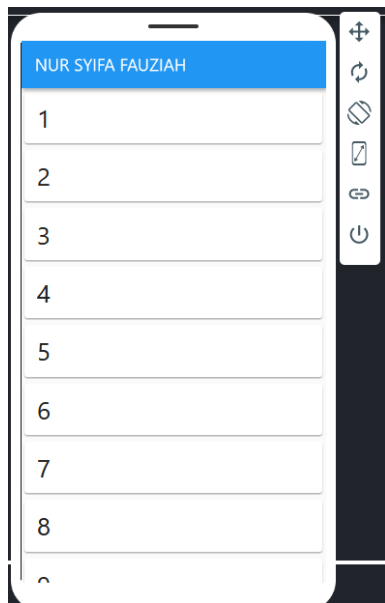
4. **Native** adalah aplikasi yang dibangun dengan bahasa pemrograman yang spesifik untuk platform tertentu. Contoh populernya yakni penggunaan bahasa pemrograman Objective-C atau Swift untuk platform iOS (Apple). Adapun platform Android yang menggunakan bahasa pemrograman Java.

- **Aplikasi hybrid** adalah aplikasi web yang ditransformasikan menjadi kode native pada platform seperti iOS atau Android. Aplikasi hybrid biasanya menggunakan browser untuk memungkinkan aplikasi web mengakses berbagai fitur di device mobile seperti Push Notification, Contacts, atau Offline Data Storage. Beberapa tools untuk mengembangkan aplikasi hybrid antara lain Phonegap, Rbymotion dan lain-lain.

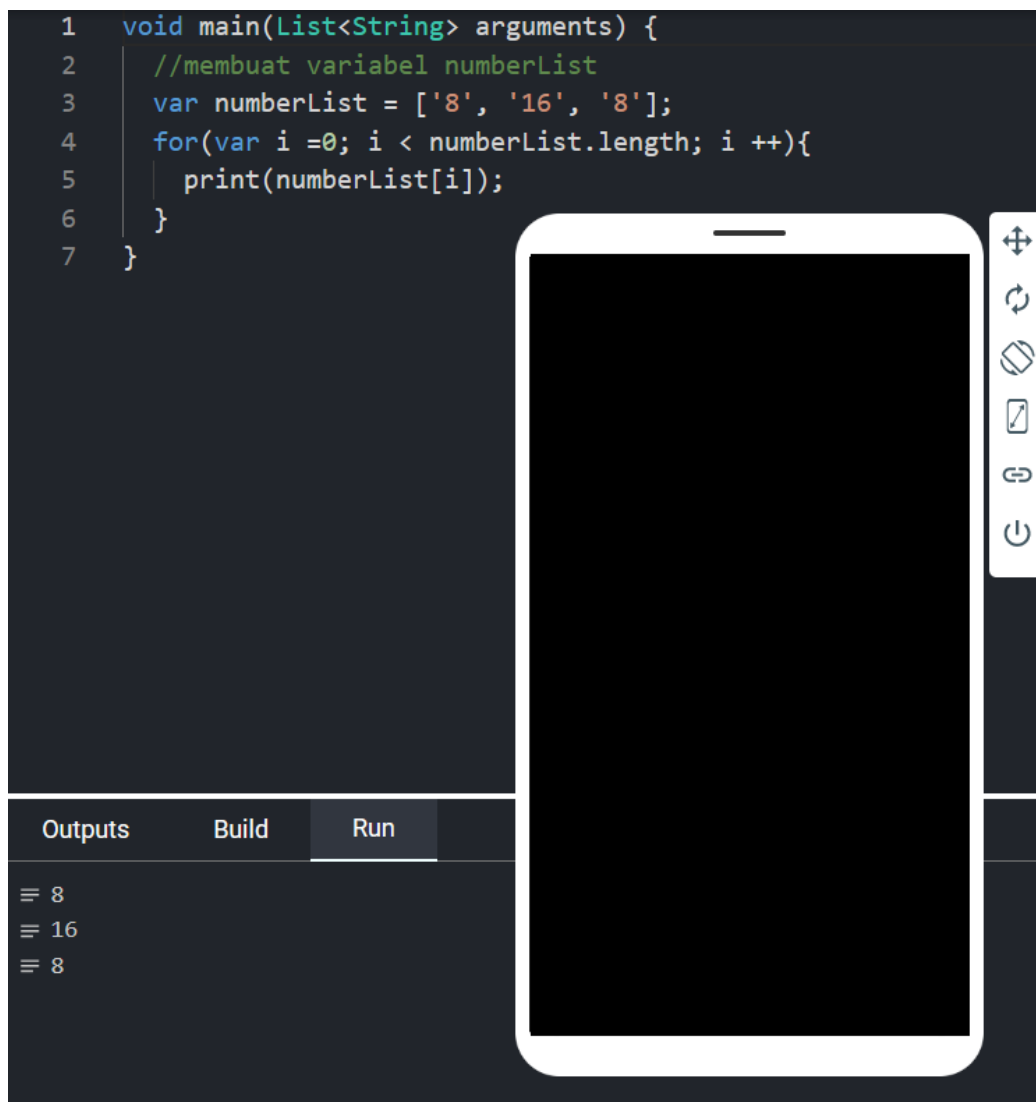
6. beberapa fungsi dari GitHub adalah antara lain:

- Dapat digunakan untuk menyimpan repository.
- Menjadi wadah atau alat untuk kolaborasi dan mengerjakan sebuah project.
- Memuat profil Anda sehingga dapat mengikuti programmer lainnya di aplikasi.
- Dapat untuk mengawasi repository tertentu.
- Merencanakan, menyimpan dan melacak proses kerja dari proyek;
- Berkomunikasi dengan sesama programmer;
- Melacak bug dan manajemen tugas.

7. Berikut hasil outputnya



8. Berikut hasil output nya



```
9. [
{
  "id": 1,
  "name": "Leanne Graham",
  "username": "Bret",
  "email": "Sincere@april.biz",
  "address": {
    "street": "Kulas Light",
    "suite": "Apt. 556", "city":
    "Gwenborough",
    "zipcode": "92998-3874",
    "geo": {
      "lat": "-37.3159",
      "lng": "81.1496"
    }
  },
  "phone": "1-770-736-8031 x56442",
  "website": "hildegard.org",
  "company": {
    "name": "Romaguera-Crona",
    "catchPhrase": "Multi-layered client-server neural-net",
    "bs": "harness real-time e-markets"
  }
},
{
  "id": 2,
  "name": "Ervin Howell",
  "username": "Antonette",
  "email": "Shanna@melissa.tv",
  "address": {
    "street": "Victor Plains",
```

```
"suite": "Suite 879",
"city": "Wisokyburgh",
"zipcode": "90566-7771",
"geo": {
  "lat": "-43.9509",
  "lng": "-34.4618"
},
"phone": "010-692-6593 x09125",
"website":
"anastasia.net",
"company": {
  "name": "Deckow-Crist",
  "catchPhrase": "Proactive didactic
contingency", "bs": "synergize scalable supply-
chains"
}
```

Tahap Pertama kita akan membuat file halamanJson.dart dalam project kita. Kemudian isi dengan kode di bawah ini.

```
import 'package:flutter/material.dart';

void main() {
  runApp(new MaterialApp(
    title: "My Apps",
    home: new HalamanJson(),
  ));
}

class HalamanJson extends StatefulWidget
{
  @override
  _HalamanJsonState createState() => _HalamanJsonState();
}
```

```

}

class _HalamanJsonState extends

State {@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: Text("Data JSON"),
    ),
    drawer: DrawerApp(),
    body: Center(
      child: Text("Data JSON")
    ),
  );
}
}

```

Selanjutnya kita membutuhkan beberapa package diantaranya sebagaiberikut. Tambahkan http dependencies dalam file pubspecs.yaml

```

dependencies:
  flutter:
    sdk: flutter

  http: ^0.12.0+1

```

Lalu import dalam file .dart

```

import 'dart:convert';
import 'package:http/http.dart' as
http;import 'dart:async';

```

Selanjutnya kita akan menggunakan **Future** untuk menjalankan `http.get`.

```
List datadariJSON;
```

```
Future ambildata() async {  
  http.Response hasil = await  
  http.get(  
    Uri.encodeFull("https://jsonplaceholder.typicode.com/users"),  
    headers: {"Accept": "application/json"});  
  
  this.setState(() {  
    datadariJSON = json.decode(hasil.body);  
  });  
}
```

Pada contoh **Future** di atas, sebelumnya kita telah membuat List terlebih dahulu yang bernama **datadariJSON** yang dimaksudkan akan menampung data yang akan diparsing dari url. Jadi sebelum Future **ambildata** dijalankan maka List **datadariJSON** masih bernilai **null**. `setState` di sini berfungsi untuk merubah state dari **datadariJSON** yang tadinya null menjadi berisi data dari hasil parsing. Lalu untuk menjalankan Future ambil data kita menggunakan **initState**.

```
@override  
void initState() {  
  this.ambildata();  
}
```

Singkatnya Full Code-nya akan menjadi seperti di bawah ini. Di sini saya menampilkan List **datadariJSON** menggunakan **ListView.builder** dimana setiap ListTile nya dapat diubah secara custom sesuai keinginan masing-masing. Untuk melihat contoh ListView custom

```
import 'dart:convert';
```



```
import 'package:flutter/material.dart';
import 'package:http/http.dart' as
http;import 'dart:async';

void main() {
  runApp(new MaterialApp(
    title: "My Apps",
    home: new HalamanJson(),
  ));
}

class HalamanJson extends StatefulWidget
{
  @override
  _HalamanJsonState createState() => _HalamanJsonState();
}

class _HalamanJsonState extends
State {List datadariJSON;

Future ambildata() async {
  http.Response hasil = await http.get(
    Uri.encodeFull("https://jsonplaceholder.typicode.com/users"),
    headers: {"Accept": "application/json"});

  this.setState(() {
    datadariJSON = json.decode(hasil.body);
  });
}

@override
```

```

void initState() {
  this.ambildata();
}

@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: Text("Data JSON"),
    ),
    body: Container(
      child: ListView.builder(
        itemCount: datadariJSON == null ? 0 :
        datadariJSON.length,itemBuilder: (context, i){
          return ListTile(
            title: Text(datadariJSON[i]['name']),
          );
        }
      ),
    ),
  );
}

```

Saat menampilkan data dari List kita seperti menampilkan array pada umumnya dan disesuaikan dengan struktur dari JSON yang tersedia.

Contohnya di sini menampilkan **name** dengan cara **datadariJSON[i]['name']**. Contoh lain jika ingin menampilkan nama jalan maka kita gunakan cara **datadariJSON[i]['address']['street']**.



YAYASAN SASMITA JAYA
UNIVERSITAS PAMULANG
DATA PEMBAYARAN SEMESTER GENAP 2020/2021
FAKULTAS / PRODI : TEKNIK / TEKNIK INFORMATIKA
NAMA MAHASISWA : NUR SYIFA FAUZIAH
NIM : 181011401055
SHIFT : REGULER C

DATA PEMBAYARAN TAGIHAN UANG KULIAH

NO	NOMOR TAGIHAN	NO URUT	PEMBAYARAN	JML BAYAR	STATUS BAYAR	TGL BAYAR	CHANNEL	TEMPAT BAYAR
1	2020113286802201	1	Registrasi	350000	LUNAS	2021-01-22 20:41:59.896	IBANKING	Bank MANDIRI
2	2020113286802301	2	SKS2	300000	LUNAS	2021-04-04 18:20:06.449	IBANKING	Bank MANDIRI
3	2020113286802401	3	SKS3	300000	LUNAS	2021-04-04 18:20:50.028	IBANKING	Bank MANDIRI
4	2020113286800501	4	UTS	350000	LUNAS	2021-04-23 21:03:44.843	IBANKING	Bank MANDIRI
5	2020113286802501	5	SKS4	300000	LUNAS	2021-05-03 22:17:50.811	IBANKING	Bank MANDIRI
6	2020113286802601	6	SKS5	300000	LUNAS	2021-05-03 22:18:50.016	IBANKING	Bank MANDIRI
7	2020113286802701	7	SKS6	300000	BELUM LUNAS			
8	2020113286800401	8	PRAKTEK	150000	BELUM LUNAS			
9	2020113286800601	9	UAS	350000	BELUM LUNAS			

DATA PEMBAYARAN TAGIHAN LAINNYA

NO	NOMOR TAGIHAN	NO URUT	PEMBAYARAN	JML BAYAR	STATUS BAYAR	TGL BAYAR	CHANNEL	TEMPAT BAYAR
----	---------------	---------	------------	-----------	--------------	-----------	---------	--------------