

Nama : Kristianto Sianipar
NIM : 171021400081

Jawaban – UAS Mobile Programming

1. Jelaskan apa yang dimaksud dengan Mobile Programming ?

Jawab :

Mobile Programming adalah suatu pemrograman yang secara singkat memiliki makna proses menulis kode-kode program untuk membuat aplikasi yang ditujukan untuk perangkat bergerak (Mobile), Mobile Programming juga bisa dikatakan gabungan antara Desktop Programming dan juga Web Programming, dikarenakan format aplikasi dari Mobile Programming yang kita kenal (apk) sendiri merupakan prototype dari Desktop Programming dan juga tampilan tampilan dari aplikasi Mobile Programming masih merupakan bagian dari Web Programming.

2. Jelaskan apa yang dimaksud dengan User Interface (UI) ?

Jawab :

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. Jelaskan apa yang dimaksud dengan API ? jelaskan fungsinya

Jawab :

Sebuah API adalah sekumpulan definisi dan protokol untuk membangun dan mengintegrasikan perangkat lunak aplikasi. API adalah singkatan dari antarmuka pemrograman aplikasi (Application Programming Interface). API memungkinkan produk atau layanan anda berkomunikasi dengan produk dan layanan lain tanpa harus tahu bagaimana penerapannya

4. Jelaskan perbedaan Native dan Hybrid pada mobile programming ?

Jawab :

Aplikasi native adalah aplikasi yang dibangun dengan bahasa pemrograman yang spesifik untuk platform tertentu. Untuk membuat aplikasi di dua sistem operasi yang berbeda, kamu membutuhkan bahasa pemrograman yang berbeda pula. Contoh populernya yakni penggunaan bahasa pemrograman Objective-C atau Swift untuk platform iOS (Apple) Sementara itu, kamu bisa menggunakan bahasa pemrograman Java untuk platform Android


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

6. Jelaskan apa fungsi github ?

Jawab :

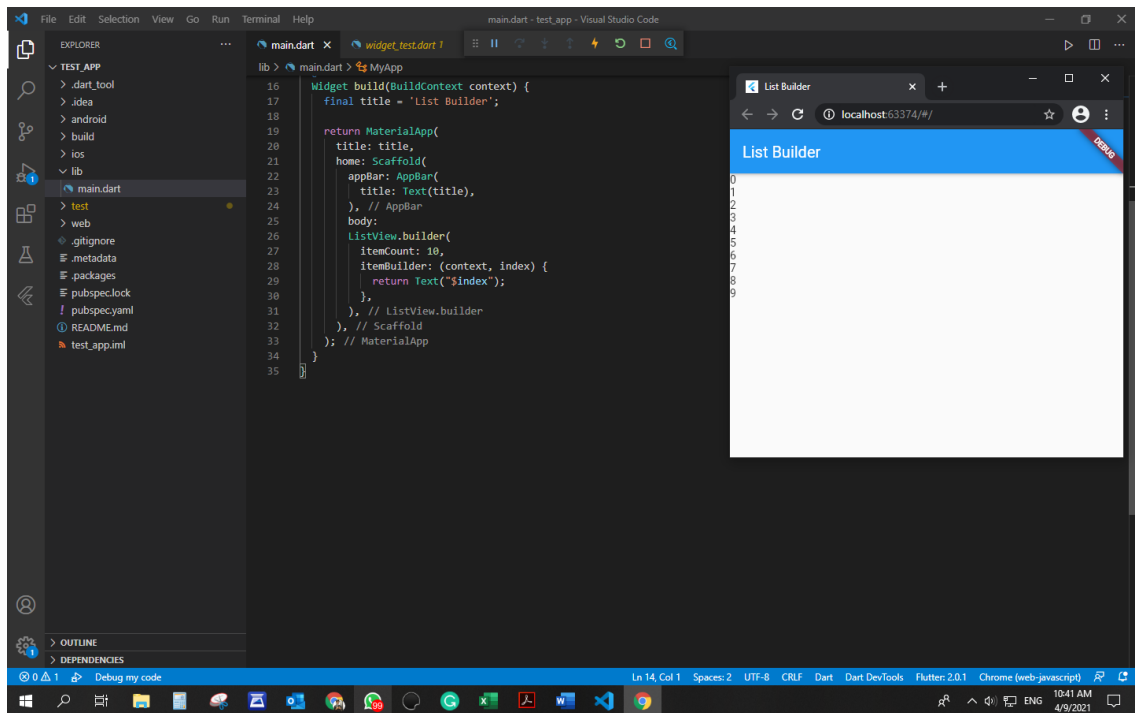
GitHub bisa dibilang adalah social media para programmer disana mereka bisa saling berbagi repo yang berisi program program yang mereka kerjakan. GitHub sendiri secara lengkap adalah manajemen proyek dan sistem versioning code sekaligus platform jaringan sosial yang dirancang khusus bagi para developer. Dengan github kita dapat bekerja bersama-sama dengan rekan dari berbagai belahan dunia, merencanakan proyek, dan bahkan tracking (melacak) pekerjaan anda.

7. Apa output dari script berikut ?



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

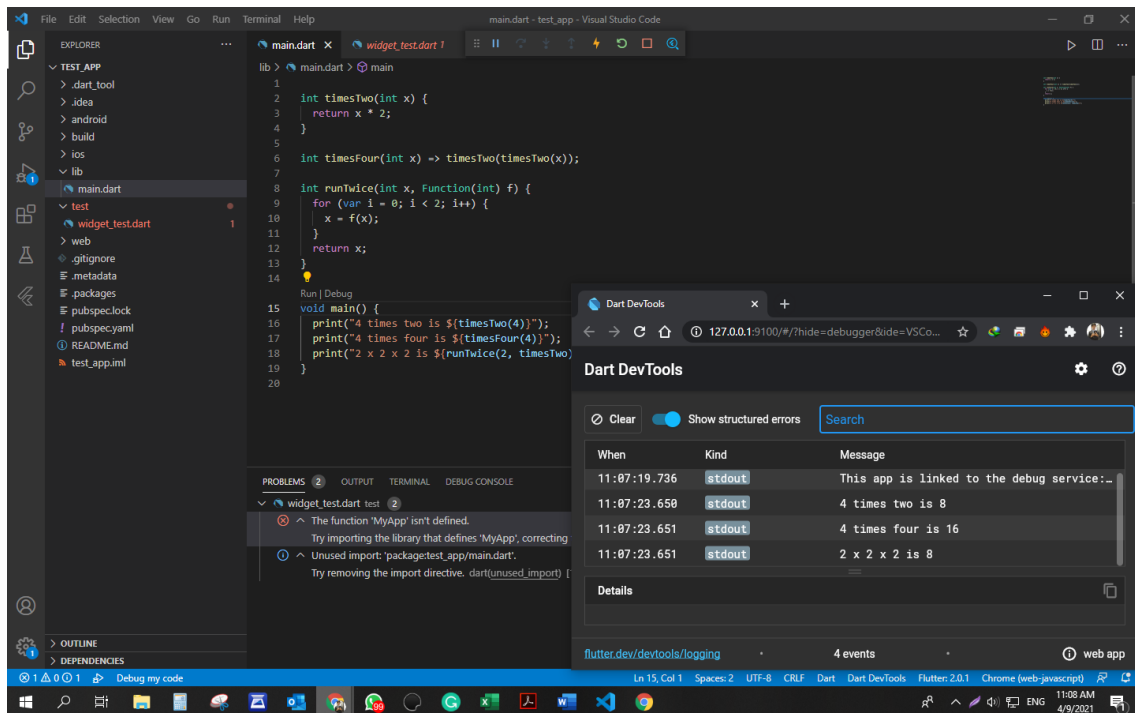
Jawaban :



8. Apa output dari script berikut ?



Jawaban :



9. Tuliskan sintak cara parsing JSON pada flutter

Jawaban :

saya akan menggunakan sampel data json dari <https://jsonplaceholder.typicode.com/users/>.

Jika dilihat struktur datanya kira-kira seperti di bawah ini.

```
[  
  {  
    "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",  
    }  
  },  
  {  
    "id": 2,  
    "name": "Ervin霍夫曼",  
    "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",  
    }  
  },  
  {  
    "id": 3,  
    "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",  
    }  
  },  
  {  
    "id": 4,  
    "name": "Ervin霍夫曼",  
    "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",  
    }  
  },  
  {  
    "id": 5,  
    "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",  
    }  
  },  
  {  
    "id": 6,  
    "name": "Ervin霍夫曼",  
    "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",  
    }  
  },  
  {  
    "id": 7,  
    "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",  
    }  
  },  
  {  
    "id": 8,  
    "name": "Ervin霍夫曼",  
    "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",  
    }  
  },  
  {  
    "id": 9,  
    "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",  
    }  
  },  
  {  
    "id": 10,  
    "name": "Ervin霍夫曼",  
    "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 halaman `Json.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 sebagai berikut. 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.

```
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']`.