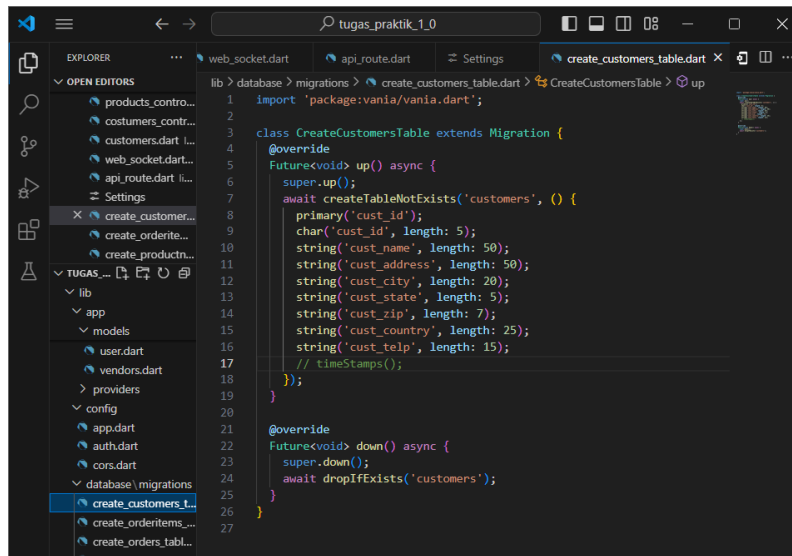**LAPORAN**
TUGAS CRUD POSTMAN

Disusun oleh:
5220411414 Muchammad Hasbi Ashshiddiqi

PROGRAM STUDI INFORMATIKA
FAKULTAS SAINS & TEKNOLOGI
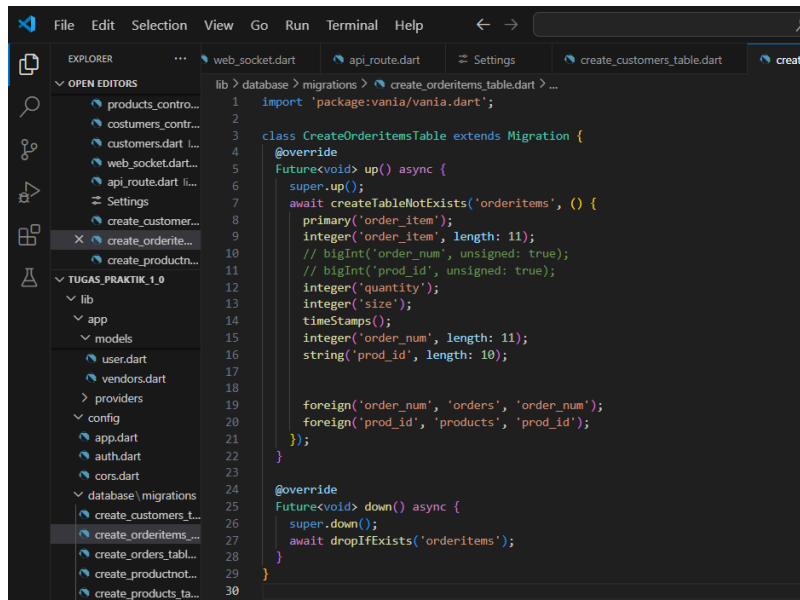**UNIVERSITAS TEKNOLOGI YOGYAKARTA**
YOGYAKARTA
2024

Source Code


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

class CreateCustomersTable extends Migration {
  @override
  Future<void> up() async {
    super.up();
    await createTableNotExists('customers', () {
      primary('cust_id');
      char('cust_id', length: 5);
      string('cust_name', length: 50);
      string('cust_address', length: 50);
      string('cust_city', length: 20);
      string('cust_state', length: 5);
      string('cust_zip', length: 7);
      string('cust_country', length: 25);
      string('cust_telp', length: 15);
      // timeStamps();
    });
  }

  @override
  Future<void> down() async {
    super.down();
    await dropIfExists('customers');
  }
}
```
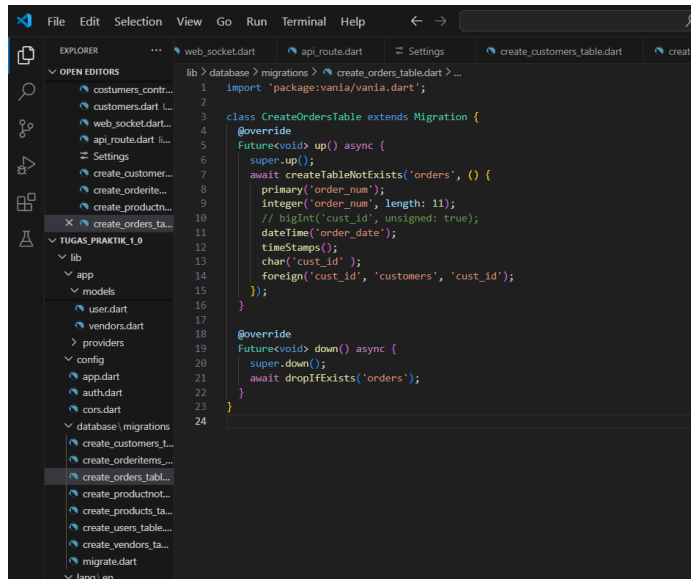


```dart
import 'package:vania/vania.dart';

class CreateOrderitemsTable extends Migration {
  @override
  Future<void> up() async {
    super.up();
    await createTableNotExists('orderitems', () {
      primary('order_item');
      integer('order_item', length: 11);
      // bigInt('order_num', unsigned: true);
      // bigInt('prod_id', unsigned: true);
      integer('quantity');
      integer('size');
      timeStamps();
      integer('order_num', length: 11);
      string('prod_id', length: 10);


      foreign('order_num', 'orders', 'order_num');
      foreign('prod_id', 'products', 'prod_id');
    });
  }

  @override
  Future<void> down() async {
    super.down();
    await dropIfExists('orderitems');
  }
}
```
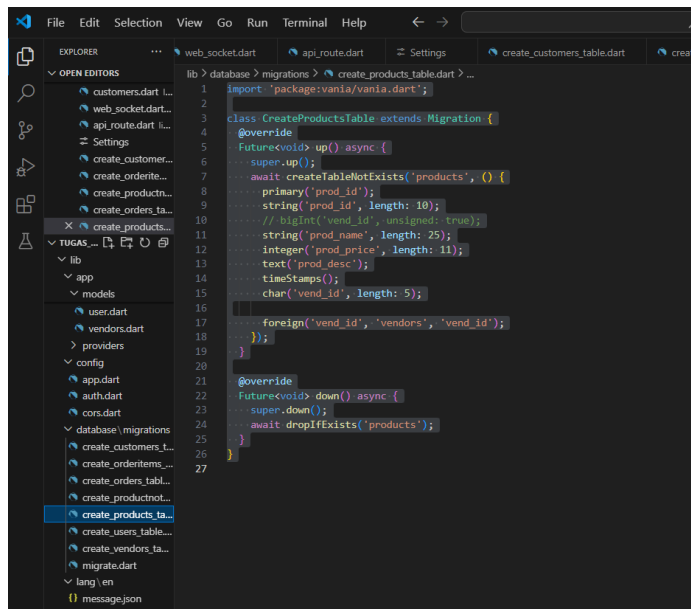
First editor — `lib > database > migrations > create_orders_table.dart`:

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

class CreateOrdersTable extends Migration {
  @override
  Future<void> up() async {
    super.up();
    await createTableNotExists('orders', () {
      primary('order_num');
      integer('order_num', length: 11);
      // bigInt('cust_id', unsigned: true);
      dateTime('order_date');
      timeStamps();
      char('cust_id' );
      foreign('cust_id', 'customers', 'cust_id');
    });
  }

  @override
  Future<void> down() async {
    super.down();
    await dropIfExists('orders');
  }
}
```

Second editor — `lib > database > migrations > create_products_table.dart`:

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

class CreateProductsTable extends Migration {
  @override
  Future<void> up() async {
    super.up();
    await createTableNotExists('products', () {
      primary('prod_id');
      string('prod_id', length: 10);
      // bigInt('vend_id', unsigned: true);
      string('prod_name', length: 25);
      integer('prod_price', length: 11);
      text('prod_desc');
      timeStamps();
      char('vend_id', length: 5);

      foreign('vend_id', 'vendors', 'vend_id');
    });
  }

  @override
  Future<void> down() async {
    super.down();
    await dropIfExists('products');
  }
}
```

```dart
lib > database > migrations > migrate.dart > Migrate > registry

import 'dart:io';
import 'package:tugas_praktik_1_0/database/migrations/create_users_table.dart';
import 'package:vania/vania.dart';
import 'create_customers_table.dart';
import 'create_orders_table.dart';
import 'create_productnotes_table.dart';
import 'create_products_table.dart';
import 'create_vendors_table.dart';
import 'create_orderitems_table.dart';


Run | Debug
void main(List<String> args) async {
  await MigrationConnection().setup();
  if (args.isNotEmpty && args.first.toLowerCase() == "migrate:fresh") {
    await Migrate().dropTables();
  } else {
    await Migrate().registry();
  }
  await MigrationConnection().closeConnection();
  exit(0);
}

class Migrate {
  registry() async {
    // await CreateUserTable().up();
    await CreateCustomersTable().up();
    await CreateOrdersTable().up();
    await CreateProductnotesTable().up();
    await CreateProductsTable().up();
    await CreateVendorsTable().up();
    await CreateOrderitemsTable().up();
  }

  dropTables() async {
    await CreateOrderitemsTable().down();
    await CreateVendorsTable().down();
    await CreateProductsTable().down();
    await CreateProductnotesTable().down();
    await CreateOrdersTable().down();
    await CreateCustomersTable().down();
    await CreateUserTable().down();
  }
}
```

```dart
lib > database > migrations > create_users_table.dart > ...

import 'package:vania/vania.dart';

class CreateUserTable extends Migration {
  @override
  Future<void> up() async {
    super.up();
    await createTableNotExists('users', () {
      id();
    });
  }

  @override
  Future<void> down() async {
    super.down();
    await dropIfExists('users');
  }
}
```

Relasi Table
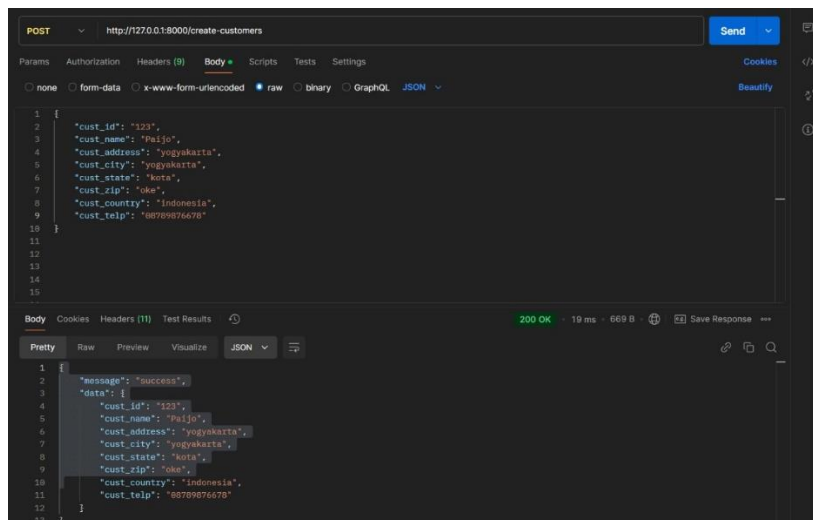


```
B:\Semester 5\Mobile Web Service Teori & Praktik\vania\tugas_praktik_1_0>vania migrate
 Migration started
 Create customers table.................................. 15ms DONE
 Create vendors table................................... 10ms DONE

B:\Semester 5\Mobile Web Service Teori & Praktik\vania\tugas_praktik_1_0>vania migrate
 Migration started
 Create customers table.................................. 3ms DONE
 Create orders table.................................... 25ms DONE
 Create products table.................................. 29ms DONE
 Create vendors table................................... 0ms DONE

B:\Semester 5\Mobile Web Service Teori & Praktik\vania\tugas_praktik_1_0>vania migrate
 Migration started
 Create customers table.................................. 2ms DONE
 Create orders table.................................... 1ms DONE
 Create productnotes table.............................. 19ms DONE
 Create products table.................................. 1ms DONE
 Create vendors table................................... 1ms DONE

B:\Semester 5\Mobile Web Service Teori & Praktik\vania\tugas_praktik_1_0>vania migrate
 Migration started
 Create customers table.................................. 2ms DONE
 Create orders table.................................... 1ms DONE
 Create productnotes table.............................. 1ms DONE
 Create products table.................................. 0ms DONE
 Create vendors table................................... 0ms DONE
 Create orderitems table................................ 29ms DONE
```

Disini hanya create

```dart
import 'package:vania/vania.dart';
import 'package:tugas_praktik_1_0/app/models/customers.dart';

class CustomersController extends Controller {
  // Mendapatkan semua pelanggan
  Future<Response> index() async {
    final customers = await Customers().query().get();
    return Response.json({'data': customers});
  }

  // Menambahkan pelanggan baru
  Future<Response> create(Request req) async {
    req.validate({
      'cust_id': 'required|string|max_length:5',
      'cust_name': 'required|string|max_length:50',
      'cust_address': 'required|string|max_length:100',
      'cust_city': 'required|string|max_length:50',
      'cust_state': 'required|string|max_length:25',
      'cust_zip': 'required|string|max_length:7',
      'cust_country': 'required|string|max_length:25',
      'cust_telp': 'required|string|max_length:15',
    }, {
      'cust_id.required': 'tidak boleh kosong',
      'cust_name.string': 'tidak boleh kosong',
      'cust_address.required': 'tidak boleh kosong',
      'cust_city.required': 'tidak boleh kosong',
      'cust_state.required': 'tidak boleh kosong',
      'cust_zip.required': 'tidak boleh kosong',
      'cust_country.required': 'tidak boleh kosong',
      'cust_telp.required': 'tidak boleh kosong',
    });

    final customerData = req.input();

    // Cek apakah ID sudah ada
    final existingCustomer = await Customers()
        .query()
        .where('cust_id', '=', customerData['cust_id'])
        .first();

    if (existingCustomer != null) {
      return Response.json(
          {'message': 'Pelanggan dengan ID ini sudah ada.'},
```

phpMyAdmin:

Menampilkan baris 0 - 0 (total 1, Pencarian dilakukan dalam 0,0003 detik.)

SELECT * FROM `customers`

cust_id	cust_name	cust_address	cust_city	cust_state	cust_zip	cust_country	cust_telp
123	Paijo	yogyakarta	yogyakarta	kota	oke	indonesia	08789876678