lib\services\api.dart

```dart
import 'dart:convert'; // Import for JSON encoding/decoding
import 'package:appwithbackend/model/product_model.dart'; // Import the Product model
import 'package:flutter/foundation.dart'; // Import for debugging
import 'package:http/http.dart'
    as http; // Import the HTTP package for making requests

class Api {
  static const baseUrl =
      "http://192.168.8.199/api/"; // Base URL of the backend API

  // POST method to add a product
  static addProduct(Map pdata) async {
    print(pdata); // Print the product data for debugging
    var url = Uri.parse("${baseUrl}add_product"); // Construct the URL

    try {
      final res = await http.post(url, body: pdata); // Send POST request

      if (res.statusCode == 200) {
        var data = jsonDecode(res.body.toString()); // Decode the response body
        print(data); // Print the response data
      } else {
        print("Failed to get response"); // Print error message
      }
    } catch (e) {
      debugPrint(e.toString()); // Print exception message
    }
  }

  // GET method to retrieve all products
  static getProduct() async {
    List<Product> products = []; // List to hold the products

    var url = Uri.parse("${baseUrl}get_product"); // Construct the URL

    try {
      final res = await http.get(url); // Send GET request
      if (res.statusCode == 200) {
        var data = jsonDecode(res.body); // Decode the response body
        for (var value in data) {
          // Iterate over the response data
          products.add(
            Product(
              name: value['pname'],
              desc: value['pdesc'],
              price: value['pprice'],
              id: value['_id'].toString(),
            ),
          );
        }
        return products; // Return the list of products
      } else {
        // Handle non-200 status code
      }
    } catch (e) {
      print(e.toString()); // Print exception message
    }
  }

  // PUT method to update a product
  static updateProduct(id, Map<String, dynamic> body) async {
    var url = Uri.parse("${baseUrl}update/$id"); // Construct the URL

    try {
      final res = await http.put(url, body: body); // Send PUT request
      if (res.statusCode == 200) {
        print(jsonDecode(res.body)); // Print the response body
      } else {
        print("Failed to update data"); // Print error message
      }
    } catch (e) {
      print(e.toString()); // Print exception message
    }
  }

  // DELETE method to delete a product
  static deleteProduct(id) async {
    var url = Uri.parse("${baseUrl}delete/$id"); // Construct the URL

    try {
      final res = await http.delete(url); // Send DELETE request
      if (res.statusCode == 204) {
        print("Product deleted"); // Print success message
      } else {
        print("Failed to delete"); // Print error message
      }
    } catch (e) {
      print(e.toString()); // Print exception message
    }
  }
}

```