# README: CoffeeShop Application

## Introduction

The CoffeeShop application simulates the basic operations of a coffee shop system. It allows adding menu items, taking orders, updating order statuses, calculating bills, and displaying active orders. This project helps understand the fundamental operations involved in managing orders within a service-oriented business.

## Project Type

Backend

## Deployed App

- Backend: \*No deployment link provided\*

## Directory Structure

```  
CoffeeShop/  
├─ coffeeshop.js  
```

## Video Walkthrough of the Project

Attach a very short video walkthrough of all of the features [1 - 3 minutes]

## Video Walkthrough of the Codebase

Attach a very short video walkthrough of the codebase [1 - 5 minutes]

## Features

* Add menu items with prices.
* Take orders and associate them with a table.
* Update the status of orders (Preparing, Served, Completed).
* Calculate the total bill for a table.
* Display active orders that are either Preparing or Served.

## Design Decisions or Assumptions

- Each table can have multiple orders, but each order can only be associated with one table.

- Orders and menu items are stored in memory; no database is used.

## Installation & Getting Started

To run the CoffeeShop application locally:  
  
```bash  
# Clone the repository (repository URL needed)  
git clone <repository\_url>  
cd CoffeeShop  
  
# There is no dependency installation or build needed, just run the script  
node coffeeshop.js  
```

## Usage

The CoffeeShop class can be used as follows:  
  
```javascript  
const coffeeShop = new CoffeeShop();  
  
// Adding menu items  
coffeeShop.addMenuItem("Espresso", 5.0);  
  
// Taking an order  
coffeeShop.takeOrder(1, ["Espresso"]);  
  
// Updating an order status  
coffeeShop.updateOrderStatus(1, "Served");  
  
// Calculating bill for a table  
coffeeShop.calculateBill(1);  
  
// Displaying active orders  
coffeeShop.displayActiveOrders();  
```