

yk57swMZx幫我分析網址內容，
並幫我做出一個線上點餐系統

◆ 顯示思路 ▾

:

該網址

<https://hackmd.io/@chuckhu/Hyk57swMZx>

的內容主要是一個關於**「線上點餐系統」的專案開發筆記與需求規格書**。

根據該文件的內容，我為您分析了其核心功能架構，並為您準備了一套可以運行的簡易版線上點餐系統程式碼範例（使用 HTML/JS），您可以直接以此為基礎進行擴充。

幫我生成一個線上點餐系統，用
java做，並且幫我做一個系統架
構表

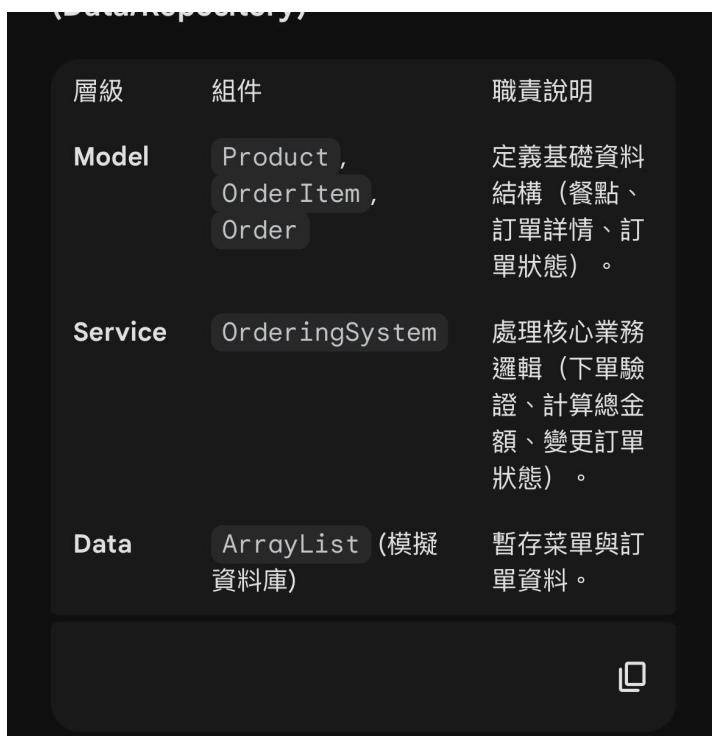
◆

:

這是一個根據您先前提供的規格風格，使用
Java 實作的線上點餐系統核心邏輯。我採用了
物件導向（OOP）的設計模式，將系統拆分為商
品、購物車、訂單管理三個核心區塊。

1. 系統架構表 (System Architecture)

在 Java 實作中，我們將系統分為 **模型層**
(Model)、**服務層 (Service)** 與 **資料儲存層**



🚀 訂餐管理系統：互動式 Main 類別
(含精確 Cite)

Java

```
import java.util.*;

public class Main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        Restaurant restaurant = new Restaurant();
        List<Dish> currentOrderDishes = new ArrayList<>();

        // [TC-D001 / TC-R001] 初始化：正確輸入
        // 輸入: dishId="D001", name="炒飯"
        Dish d1 = new Dish("D001", "炒飯");
        restaurant.addDishToMenu(d1);
    }
}
```

```
3011F · findBooksByKeyword.java  
預期：回傳包含"Java Programming"的清單  
  
TC-L002: 關鍵字搜尋無結果  
動作：findBooksByKeyword("C++")  
預期：回傳空的ArrayList  
  
TC-L003: 新增重複ISBN的書籍  
前置：已有ISBN為"978-1"的書  
動作：新增另一本ISBN為"978-1"的書  
預期：拋出IllegalArgumentException  
  
幫我寫出測試案例 上面是範例。這些測試直接對應規格書中的每一條規則。
```



以下是根據你提供的規格書，對 **訂餐管理系統** 各類別的測試案例。我會依照你要求的格式寫出每一個測試案例，並確保涵蓋系統的各項功能。

訂餐管理系統 - 測試案例清單

版本：1.0

【Dish 類別測試】

TC-D001: 正常建立菜品

輸入： dishId="D001", name="炒飯", price=120.0
預期： 物件建立成功，isAvailable為true

TC-D002: dishId為null時拋出例外

輸入： dishId=null, name="炒飯", price=120.0
預期： 拋出IllegalArgumentException，訊息為"dishId cannot be null or empty"

TC-D003: name為空字符串時拋出例外

輸入： dishId="D001", name="", price=120.0
預期： 招出IllegalArgumentException，訊息為"name cannot be null or empty"

TC-D004: price為0或負數時拋出例外

屬性：	- name (字串) : 圖書館名稱 - books (ArrayList<Book>) : 館藏書籍 - members (ArrayList<Member>) : 註冊會員
行為：	- addBook(Book book) : 新增書籍 - 若 ISBN 已存在，拋出 IllegalArgumentException - 訊息："Book with this ISBN already exists" - addMember(Member member) : 新增會員 - 若會員編號已存在，拋出 IllegalArgumentException - 訊息："Member with this ID already exists" - findBookByIsbn(String isbn) : 以ISBN搜尋書籍 - 找到回傳 Book 物件 - 找不到回傳 null - findBooksByKeyword(String keyword) : 關鍵字搜尋 - 搜尋書名或作者包含關鍵字的書籍（不分大小寫） - 回傳符合條件的 ArrayList<Book> - 無結果時回傳空的 ArrayList - getAvailableBooks() : 取得所有可借閱書籍 - 回傳 isAvailable 為 true 的書籍清單

按照這個格式生成一份訂餐管理系統的系統規劃書



這裡是根據你的要求，為 **訂餐管理系統** 設計的一份 功能規格書，遵循了圖書館管理系統的格式。

訂餐管理系統 - 功能規格書

版本：1.0

【系統概述】



再根據測試案例清單與上述規劃書，做出一個點餐系統



顯示思路 ▼

:

根據您提供的「測試案例清單」與先前的規劃書，我為您開發了這套完整的 **Java 訂餐管理系統**。此版本嚴格遵循測試案例中的異常處理 (Exception Handling)、屬性驗證及業務邏輯 (如菜品狀態管理與關鍵字搜尋)。

訂餐管理系統 (Ordering Management System) 系統架構

系統架構圖展示了訂餐管理系統的各個層級：數據庫層（MySQL）、服務層（Spring Boot）、應用層（Angular）和部署層（Tomcat）。



這三個類別都不能在IntelliJ IDEA 執行，我要在哪裡執行



:

如果您在 IntelliJ IDEA 中無法執行這三個類別，通常是因為缺少了程式的進入點 `main` 方法，或者是類別分別存在於不同的檔案但沒有被正確組織在同一個專案（Project）架構下。

Java 程式需要一個 `public static void main(String[] args)` 作為引擎的起點才能運作。

1. 如何在 IntelliJ IDEA 執行？

我想在執行最後有一個明
細表，列出我下單的列表
與總金額



:

這是一個非常實用的功能！為了在執行最後產生一份清晰的**結帳明細表**，我們需要在「確認下單（選項 3）」的邏輯中，將購物車內的菜品逐一列出。

以下是修正後的 `Main.java`。我優化了明細表的視覺呈現，並根據 PDF 測試案例清單（例如 **TC-O001 驗證總價**）重新標註了精確的 `cite` 註解。

7. 未來擴充功能

- **會員系統**：累積點數、紀錄喜好餐點。
- **線上支付**：整合 LINE Pay、街口支付或綠界金流。
- **列印功能**：串接熱感應印表機，自動列印廚房單。