

一、構想：

根據抽到的主題與老師提供的參考說明進行改寫與優化，使整個線上訂餐系統看起來更完善及豐富程度高。

二、系統規格書生成過程：

=====

線上訂餐系統(Online Food Ordering System)- 功能規格書
版本: 1.0

=====

【系統概述】

一個模擬 Uber Eats 或 foodpanda 核心流程的線上訂餐系統，可管理餐廳菜單、餐點客製化、訂單處理、折扣與外送費計算。

系統設計重點在於不同餐點類型的計價方式、套餐遞迴計算，以及折扣與準備時間的整合。

【類別規格】

抽象類別: MenuItem(菜單項目)

屬性:

- name(字串): 餐點名稱，不可為空或 null
- basePrice(double): 餐點基礎價格，不可為負數，必須大於 0

行為:

- 建構子 MenuItem(String name, double basePrice)
 - 若 name 為 null 或空字串，拋出 IllegalArgumentException
 - 訊息："name cannot be null or empty"
 - 若 basePrice ≤ 0，拋出 IllegalArgumentException
 - 訊息："basePrice must be greater than 0"
- calculatePrice(): double(抽象方法)
 - 由子類別實作
 - 回傳該餐點計算後的最終價格
- toString()
 - 回傳格式："name - price"

類別 : MainDish(主餐)

繼承 :

- MenuItem(菜單項目)

屬性 :

- size(字串) : 份量選項 (Small / Medium / Large)

行為 :

- 建構子 MainDish(String name, double basePrice, String size)

- 驗證規則同 MenuItem(菜單項目)

- 若 size 非合法選項, 拋出 IllegalArgumentException

- 訊息 :"Invalid size option"

- 不同 size 會影響最終價格

- calculatePrice(): double

- 根據 size 調整價格後回傳結果

- Small: 原價

- Medium: +20

- Large: +40

類別 : SideDish(附餐)

繼承 :

- MenuItem(菜單項目)

屬性 :

- upgraded(布林) : 是否升級附餐, 預設為 false

行為 :

- 建構子 SideDish(String name, double basePrice)

- 驗證規則同 MenuItem

- upgrade()

- 將 upgraded 設為 true
 - calculatePrice(): double
 - 若 upgraded 為 false, 回傳 basePrice
 - 若 upgraded 為 true, 回傳 basePrice 加升級費用(+10)
-

類別 : Beverage(飲料)

繼承 :

- MenuItem(菜單項目)
- 實作 Customizable 介面

屬性 :

- customizations(ArrayList<String>): 冰塊選項(去冰 / 少冰 / 正常冰)

行為 :

- 建構子 Beverage(String name, double basePrice)
 - 驗證規則同 MenuItem
 - 初始化 customizations 為空清單
 - applyCustomization(String option)
 - 將客製化選項加入清單
 - 若 option 為 null 或空字串, 拋出 IllegalArgumentException
 - 將 option 加入 customizations
 - getCustomizations()
 - 回傳所有客製化選項
 - calculatePrice(): double
 - 回傳 basePrice
-

類別 : ComboMeal(套餐)

繼承 :

- MenuItem(菜單項目)

屬性：

- items(ArrayList<MenuItem>) : 套餐內含餐點
- discountRate(double) : 套餐折扣率, 固定為 0.85

行為：

- 建構子 ComboMeal(String name)
 - 初始化 items 為空清單
 - 設定 discountRate 為 0.85
- addItem(MenuItem item)
 - 若 item 為 null, 拋出 IllegalArgumentException
 - 訊息："MenuItem cannot be null"
 - 將 item (餐點)加入套餐
- calculatePrice(): double
 - 加總所有內含 MenuItem 價格(遞迴計算)
 - 套用套餐折扣(85 折)
 - 套餐內之餐點不再套用個別折扣
 - 回傳四捨五入後的整數價格

介面：Discountable(可折扣項目)

行為：

- calculateDiscount(DiscountType type): double
 - 回傳折扣金額
- isDiscountApplicable(DiscountType type): boolean
 - 判斷是否可套用指定折扣

介面：Customizable(可客製化項目)

行為：

- applyCustomization(String option): void
- getCustomizations(): ArrayList<String>

- getCustomizationFee(): double

類別: Restaurant(餐廳)

屬性:

- name(字串): 餐廳名稱
- menu(ArrayList<MenuItem>): 餐廳菜單項目

行為:

- 建構子 Restaurant(String name)
 - 若 name 為 null 或空字串, 拋出 IllegalArgumentException
 - 訊息: "Menu item already exists"
- addMenuItem(MenuItem item)
 - 若菜單中已有相同名稱餐點, 拋出 IllegalArgumentException
 - 訊息: "Menu item already exists"
- getMenu()
 - 回傳所有菜單項目

類別: Order(訂單)

屬性:

- items(ArrayList<MenuItem>): 訂單餐點
- customer(Customer): 下單顧客

行為:

- addItem(MenuItem item)
 - 若 item 為 null, 拋出 IllegalArgumentException
 - 將餐點加入訂單
- calculateTotalPrice(): double
 - 加總所有餐點價格
 - 訊套用折扣邏輯

- 回傳最終金額
-

類別 : Customer(顧客)

屬性 :

- customerId(字串) : 顧客編號, 不可為空或 null
- name(字串) : 顧客姓名, 不可為空或 null

行為 :

- 建構子 Customer(String customerId, String name)
 - 判斷顧客是否符合折扣條件
-

類別 : DeliveryFeeCalculator(外送費計算器)

行為 :

- calculateFee(double distance, boolean isPeakTime) : double
- 依距離計算基本外送費
- 尖峰時段加收額外費用
- 回傳最終外送費金額

