

測試案例清單

【MenuItem 抽象類別測試】

TC-MI001: 名稱為 null 時拋出例外

輸入: name=null

預期: 拋出 IllegalArgumentException, 訊息為 "name cannot be null or empty"

TC-MI002: 名稱為空字串時拋出例外

輸入: name=""

預期: 拋出 IllegalArgumentException, 訊息為 "name cannot be null or empty"

TC-MI003: 價格小於 0 時拋出例外

輸入: basePrice=-1

預期: 拋出 IllegalArgumentException, 訊息為 "basePrice cannot be negative"

TC-MI004: 正常建立菜單項目

輸入: name="chickenRice", basePrice=120

預期: 物件建立成功, getName() 與 getBasePrice() 回傳正確數值

【MainDish 類別測試】

TC-MD001: 指定無效份量選項拋出例外

輸入: size="XL"

預期: 拋出 IllegalArgumentException, 訊息為 "Invalid size option"

TC-MD002: Small 份量價格

輸入: basePrice=100, size="Small"

預期: calculatePrice() 回傳 100.0

TC-MD003: Medium 份量加價 20

輸入: basePrice=100, size="Medium"

預期: calculatePrice() 回傳 120.0

TC-MD004: Large 份量加價 40

輸入: basePrice=100, size="Large"

預期: calculatePrice() 回傳 140.0

【SideDish 類別測試】

TC-SD001: 未升級附餐價格為基礎價格

輸入: basePrice=40

預期: calculatePrice() 回傳 40

TC-SD002: 升級附餐後加價 10

前置: 呼叫 upgrade()

預期: calculatePrice() 回傳 50

【Beverage 類別測試】

TC-BV001: 客製化增加成功並加入清單

動作: applyCustomization("去冰")

預期: customizations.size() 增加 1

TC-BV002: 客製化不影響價格

前置: 加入多個客製化選項

預期: getCustomizationFee() 回傳 0.0, calculatePrice() 等於 basePrice

【ComboMeal 類別測試】

TC-CM001: 套餐價格套用 85 折

前置: 放入一個 100 元與一個 50 元餐點(總計 150)

預期: calculatePrice() 四捨五入後回傳 128.0 (計算: $150 * 0.85 = 127.5$, Math.round 後為 128)

TC-CM002: 套餐名稱初始化

前置: name="超值全餐"

預期: getBasePrice() 為 0.0 (依據建構子 super(name, 0))

【Restaurant 類別測試】

TC-RS001: 餐廳名稱為空字串拋出例外

輸入: name=""

預期: 拋出 IllegalArgumentException, 訊息為 "Restaurant name cannot be empty"

TC-RS002: 新增重複名稱餐點拋出例外

前置: 菜單已有 name="cola" 的餐點

動作: addItem(new Beverage("cola", 30))

預期: 拋出 IllegalArgumentException, 訊息為 "Menu item already exists"

【Order 類別測試】

TC-OR001: 計算訂單總價(多型測試)

前置: 加入 MainDish(Large) 140元 + SideDish(Upgraded) 50元

預期: calculateTotalPrice() 回傳 190.0

【Customer 顧客類別測試】

TC-CS001: 顧客編號為空字串拋出例外

輸入: customerId=""

預期: 拋出 IllegalArgumentException

TC-CS002: 顧客姓名為空字串拋出例外

輸入: name=""

預期: 拋出 IllegalArgumentException

TC-CS003: 正常建立顧客物件

輸入: customerId="C001", name="Mary"

預期: 物件建立成功

【DeliveryFeeCalculator 外送費測試】

TC-DF001: 非尖峰時段外送費計算

輸入: distance=5, isPeakTime=false

預期: 回傳 50.0 (距離 * 10)

TC-DF002: 尖峰時段外送費加成

輸入: distance=5, isPeakTime=true

預期: 回傳 70.0 (50 + 20)