A blue and black logo

AI-generated content may be incorrect.

**โครงงานเรื่อง โปรเเกรมจำลองการทำงานของเครื่องจำหน่ายสินค้าอัตโนมัติ**

**(Vending Machine )**

**คณะผู้จัดทำโครงงาน**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 6810301002 | นางสาว | นรภร | มุตติศานต์ | **Section(S08)  (ผู้ประสานงาน)** |
| 6810301007 | นางสาว | วิภาวี | พรมสุข | **Section(S08)** |
| 6810301018 | นาย | เกียรติศักดิ์ | ปรีชานนท์ | **Section(S08)** |
| 6810301039 | นาย | ณัฐดนัย | สายสกุล | **Section(S08)** |
| 6810301042 | นางสาว | กรัณฑ์รัตน์ | จิตกิจติจำรัส | **Section(S08)** |

**เสนอ**

ผู้ช่วยศาสตราจารย์มานพ คงคานิธิ

รายงานนี้ส่วนหนึ่งของรายวิชาพื้นฐานการเขียนโปรเเกรม รหัสวิชา 300-1104

ภาคการศึกษาปีที่ 1 ปีการศึกษา 2568 สถาบันเทคโนโลยีจิตรลดา

**คำนำ**

รายงานฉบับนี้เป็นส่วนหนึ่งของรายวิชาพื้นฐานการเขียนโปรเเกรม (รหัสวิชา 300-1104) ระดับชั้นอุดมศึกษาปีที่ 1 โดยมีจุดประสงค์ เพื่อการศึกษาความรู้ด้านการเขียนโปรแกรมขั้นพื้นฐาน ซึ่งรายงานนี้มีเนื้อหาเกี่ยวกับการเขียนโปรแกรมจำลองการทำงานของเครื่องจำหน่ายสินค้าอัตโนมัติ (Vending Machine) ผู้จัดทำได้เลือกหัวข้อนี้ในการทำรายงาน เนื่องมาจากเป็นเรื่องที่น่าสนใจ ผู้จัดทำจะต้องขอขอบคุณ คณะอาจารย์ผู้ให้ความรู้ และแนวทางการศึกษา เพื่อนๆ ทุกคนที่ให้ความช่วยเหลือมาโดยตลอด

ผู้จัดทำคาดหวังเป็นอย่างยิ่งว่าการจัดทำรายงานฉบับนี้จะมีความรู้ที่เป็นประโยชน์ต่อผู้ที่สนใจศึกษาการเขียนโปรแกรมเป็นอย่างดี

**สารบัญ**

[**ข้อกำหนด 3**](#_Toc208932931)

[**คู่มือการใช้งาน 4**](#_Toc208932932)**-9**

[**Program Function 10**](#_Toc208932933)**-11**

**Flowchart 12-17**

# **ข้อกำหนด**

1. ขายสินค้าได้ 9 รายการ ราคาขายต่อหน่วย ให้เป็นเลขจำนวนเต็ม มีราคาตั้งแต่ 10 ถึง 100 เช่น 10, 15, 22, 45, 85 เป็นต้น
2. มีการควบคุมจำนวนสินค้าคงเหลือในแต่ละชนิด เพื่อสามารถขายให้แก่ลูกค้าได้ตามต้องการ

**3.** การชำระสินค้าทำได้โดยใช้เงินสด และเครื่องสามารถรับเหรียญกษาปณ์ราคา 1, 2, 5 และ 10 บาท หรือธนบัตร 20, 50, 100, 500 และ 1000 บาท ได้

**4.** มีระบบควบคุมจำนวนเหรียญกษาปณ์ และ ธนบัตรแต่ละชนิดราคา เพื่อใช้ในการทอนเงิน ให้แก่ลูกค้า และใช้เก็บเป็นเงินนำส่ง

**5 .** การซื้อสินค้าทำได้ครั้งละ หนึ่งชนิด, ชนิดละ หนึ่งชิ้น เท่านั้น

**6.** การใส่เงินชำระค่าสินค้า ทำได้ครั้งละหนึ่งชนิดเงิน โดยพิมพ์ชนิดเงิน(ค่าเงิน) เป็นตัวเลขทางคีย์บอร์ด ทำได้หลายครั้ง จนกว่ามูลค่าของเงินสดที่ได้มีค่า มากกว่าหรือเท่ากับ ราคาสินค้า

**7.** การจ่ายสินค้า และทอนเงินทำโดยการแสดง ชื่อสินค้า, ราคาสินค้า, เงินสดรวมที่ชำระ และเงินทอน โดยแยกเป็นชนิดเงิน-จำนวน

**8.** การขายไม่สำเร็จเกิดจาก กรณีต่าง เช่น ไม่มีสินค้า, ไม่มีเงินทอน, อื่นๆ เป็นต้น ระบบต้องแสดง

สถานะขายไม่สำเร็จ เงินสดรวมที่ชำระ(ถ้ามี) และจ่ายเงินสดทั้งหมดคืน(ถ้ามี) โดยแยกเป็นชนิดเงิน-จำนวน

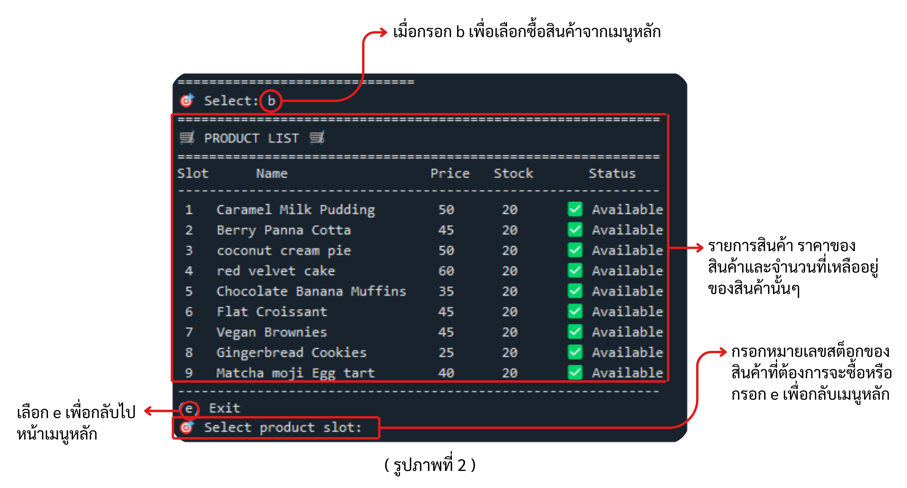
## **คู่มือการใช้งาน**

**รายละเอียดโปรเจกต์**

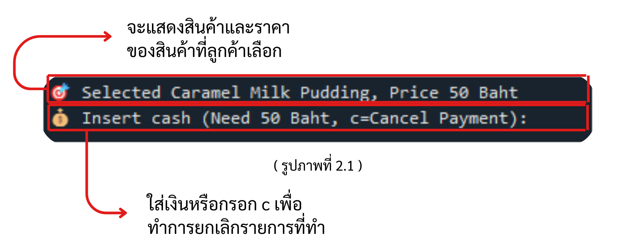
โปรเจกต์นี้เป็น โปรแกรมจำลองเครื่องขายสินค้าอัตโนมัติ (Vending Machine) เขียนด้วยภาษา Python 3 รองรับการทำงาน 2 ส่วนหลัก:

**โหมดผู้ใช้งานทั่วไป**

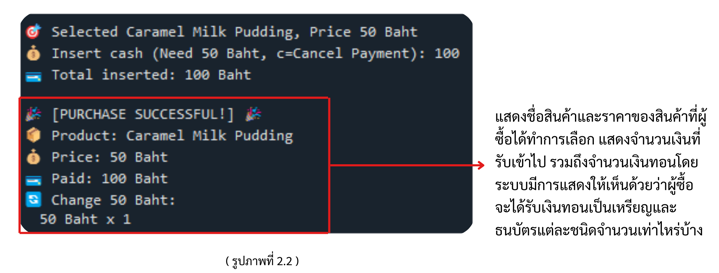
* เลือกซื้อสินค้า



* หยอดเหรียญ/ธนบัตร



* รับสินค้าและเงินทอน



**โหมดผู้ดูแล (Maintenance)**

* ตั้งค่าสินค้า 9 ช่อง
* ตั้งค่าเงินเหรียญ/ธนบัตรในเครื่อง
* ปิดระบบอย่างปลอดภัย

**คุณสมบัติหลัก**

* แสดงสินค้าพร้อมราคาและจำนวนคงเหลือ
* ตรวจสอบเงินทอนอัตโนมัติ (ถ้าไม่พอ → ยกเลิกการขาย)
* รองรับเหรียญ 1, 2, 5, 10 และ ธนบัตร 20, 50, 100, 500, 1000 บาท
* บันทึกข้อมูลสินค้าและเงินในไฟล์ (Goods.txt, Wallet.txt)
* ระบบรักษาความปลอดภัยด้วยรหัสผ่าน (ค่าเริ่มต้นคือ 1234)

**โครงสร้างไฟล์**

G08\_TermProject/

│── cmd\_G08\_TP.bat # ทำหน้าที่เป็น command run ผ่าน Terminal

│── mainprog.py    # โปรแกรมหลัก

│── storage.py   # จัดการสินค้า ไฟล์และเงิน

│── vending.py   # การจัดการคลาสและฟังชั่นเมนูต่างๆ

│── Goods.txt      # เก็บข้อมูลสินค้า

│── Wallet.txt      # เก็บข้อมูลเงินในเครื่อง

│── G08\_TP\_Report.pdf #เล่มรายงาน

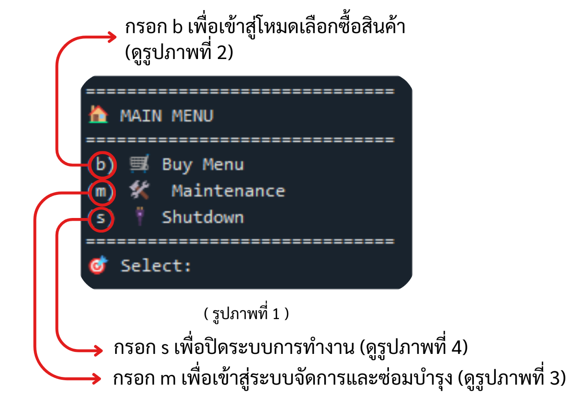
│── G08\_TP\_Report.docx #เล่มรายงาน

**วิธีการใช้งาน**

1. ดาวโหลดไฟล์ก่อนใช้งาน (ลิงค์:[G08\_TermProject.zip](https://cdti365-my.sharepoint.com/:u:/g/personal/6810301039_ms_cdti_ac_th/EY_0Bbl7hdlOki9yYvyucOcBrguhPZKPRvoF-dJrSwJIDQ?e=dd1CfY) )
2. แตกไฟล์ G08\_TermProject.zip ให้เรียบร้อย
3. รันโปรแกรมโดยรันด้วยไฟล์ batch โดยการดับเบิลคลิกไฟล์ชื่อว่า cmd\_G08\_TP.bat

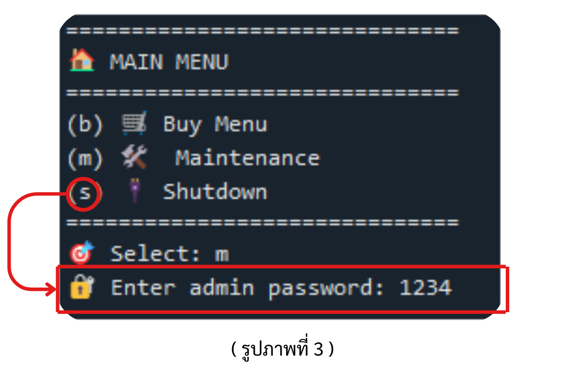
**4.** เมื่อโปรแกรมเริ่มทำงาน → เลือกโหมดการทำงาน

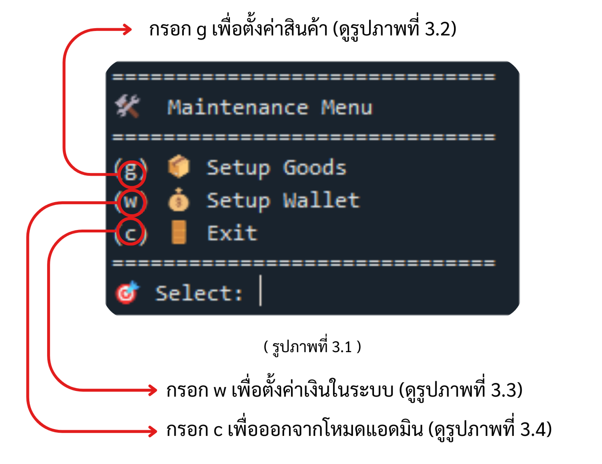
* (b) Buy Menu → สำหรับซื้อสินค้า
* (m) Maintenance → สำหรับผู้ดูแล (ใช้รหัสผ่าน)
* (s) Shutdown → ปิดระบบ

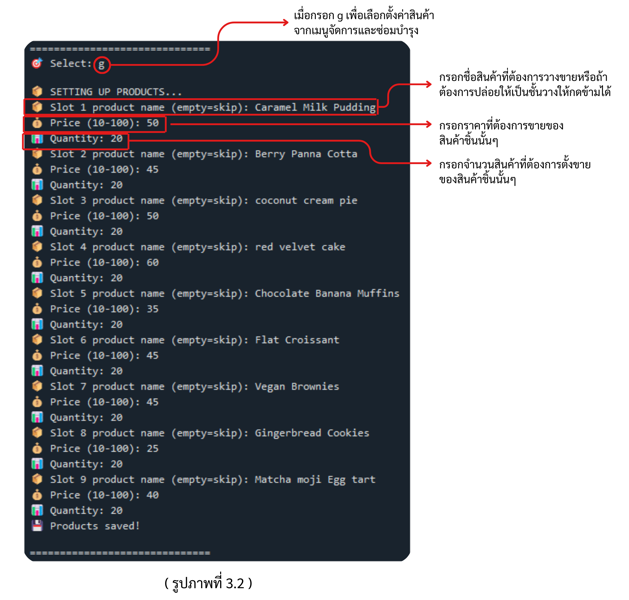


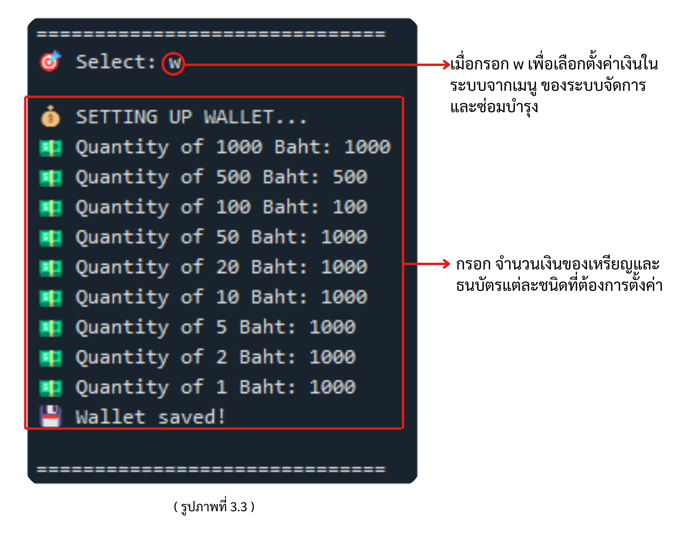
**การเข้าโหมดผู้ดูแล**

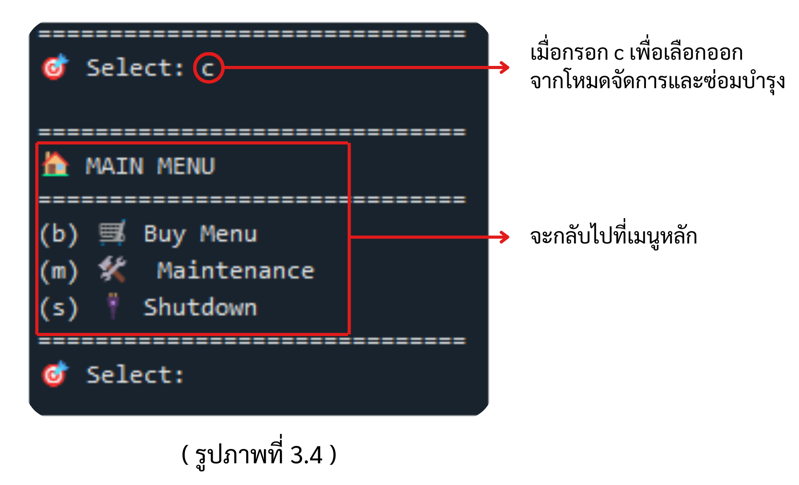
* รหัสผ่านค่าเริ่มต้น: **1234**
* สามารถเปลี่ยนสินค้า / ราคา / สต็อก และจำนวนเงินในเครื่องได้



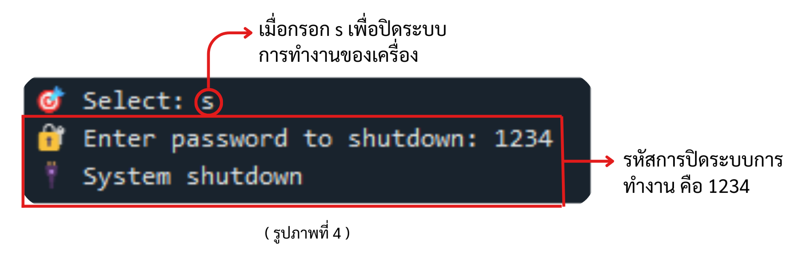








* การปิดระบบการทำงาน



### **Program Function**

ฟังก์ชันและเมธอดที่ใช้ในโปรแกรมสามารถแบ่งออกได้ดังนี้

**1) Class Product**

* **\_\_init\_\_(name, price, stock)**   
  ใช้สำหรับสร้างวัตถุสินค้า กำหนดชื่อสินค้า ราคา และจำนวนคงเหลือ
* **\_\_str\_\_()**   
  แสดงข้อความสรุปข้อมูลสินค้า (ชื่อ, ราคา, จำนวนคงเหลือ)
* **is\_available()**   
  ตรวจสอบว่าสินค้ามีคงเหลือในสต็อกหรือไม่ (True/False)
* **buy()**   
  ลดจำนวนสินค้าคงเหลือลง 1 ชิ้น เมื่อการซื้อสำเร็จ

**2) Class CashMgr**

* **\_\_init\_\_()**   
  สร้างตัวจัดการเงินสด และเก็บข้อมูลจำนวนเหรียญ/ธนบัตร
* **add(denom, count)**   
  เพิ่มจำนวนเงิน (ธนบัตร/เหรียญ) เข้ามาในเครื่อง
* **make\_change(amt)**   
  คำนวณการทอนเงินจากเงินที่มีอยู่ในเครื่อง โดยเลือกใช้เหรียญ/ธนบัตรที่เหมาะสม หากไม่สามารถทอนได้ คืนค่า None

**3) Class FileMgr**

* **load\_goods(file="Goods.txt")**   
  โหลดข้อมูลสินค้า (ชื่อ, ราคา, จำนวนคงเหลือ) จากไฟล์ Goods.txt
* **save\_goods(products, file="Goods.txt")**   
  บันทึกข้อมูลสินค้าล่าสุดกลับไปที่ไฟล์ Goods.txt
* **load\_wallet(file="Wallet.txt")**   
  โหลดข้อมูลจำนวนเหรียญและธนบัตรในเครื่องจากไฟล์ Wallet.txt
* **save\_wallet(cash, file="Wallet.txt")**   
  บันทึกข้อมูลจำนวนเหรียญและธนบัตรล่าสุดกลับไปที่ไฟล์ Wallet.txt

**4) Class VendingMachine**

* **\_\_init\_\_(password="1234")**   
  สร้างเครื่องขายสินค้าอัตโนมัติ โดยโหลดข้อมูลสินค้าและเงินจากไฟล์
* **format\_cash(cash)**   
  แปลงข้อมูลเงินให้อยู่ในรูปแบบข้อความ (ชนิดเงิน x จำนวน)
* **show\_products()**   
  แสดงรายการสินค้าในช่อง 1–9 พร้อมราคาและจำนวนคงเหลือ
* **buy\_menu()**   
  เมนูสำหรับลูกค้าเลือกซื้อสินค้า
* **collect\_cash(product)**

ทำหน้าที่ รวบรวมและตรวจสอบเงินที่ผู้ใช้จ่าย เพื่อซื้อสินค้า

* **process\_purchase(product)**   
  กระบวนการซื้อสินค้า → รับเงิน, ตรวจสอบการชำระ, ทอนเงิน, ตัด stock, บันทึกไฟล์
* **maintenance()**   
  เข้าสู่ระบบซ่อมบำรุง (ต้องใส่รหัสผ่าน) สำหรับตั้งค่าสินค้าและเงินในเครื่อง
* **setup\_goods()**   
  จัดสินค้าใหม่ในตู้ โดยให้ผู้ใช้ป้อนชื่อ, ราคา, และจำนวนสินค้า
* **setup\_wallet()**   
  กำหนดจำนวนเหรียญ/ธนบัตรใหม่ในเครื่อง
* **shutdown()**   
  ปิดระบบ (ต้องใส่รหัสผ่านก่อน)
* **run()**   
  เมนูหลักของโปรแกรม มี 3 ตัวเลือก: (b) ซื้อสินค้า, (m) ระบบซ่อมบำรุง, (s) ปิดระบบ

**Flowchart**

