



20 years old have a private mountain.

อายุ 20 มีดอยส่วนตัว

โดย

นายอิษย์ นิยมสันติ 1640703870 Section 127 B

อาจารย์ผู้สอน

ผู้ช่วยศาสตราจารย์สิรินธร จิยาศักดิ์

หลักสูตรวิทยาศาสตรบัณฑิต

ภาคการศึกษาที่ 1 ปีการศึกษา 2564

ภาควิชาวิทยาการคอมพิวเตอร์

คณะเทคโนโลยีสารสนเทศและนวัตกรรม

มหาวิทยาลัยกรุงเทพ

สารบัญ	หน้า
บทที่	
1. บทนำ	3
1.1 ชื่อโครงการ	3
1.2 วัตถุประสงค์ของโครงการ	3
1.3 ขอบเขตของโครงการ	3
2. การทำงานของแอปพลิเคชันหรือโปรแกรม	5
2.1 แผนภาพแสดงกระบวนการทำงานของโปรแกรม (Flowchart)	5
2.2 ฟังก์ชันการทำงานเริ่มต้นโปรแกรม	7
2.3 ฟังก์ชันการซื้อ	7
2.4 ฟังก์ชันเมนูขาย	8
2.5 ฟังก์ชันขายแบบเปอร์เซ็นต์	9
2.6 ฟังก์ชันขายแบบจำนวนเงินที่ต้องการรับ	10
2.7 ฟังก์ชันขายแบบจำนวนหุ้น	11
2.8 ฟังก์ชันการแสดงผล	12
2.9 ฟังก์ชันเมนูสุดท้าย	13
2.10 ฟังก์ชันซื้อเพิ่ม	14
2.11 ฟังก์ชันลบข้อมูลทั้งหมด	15
2.12 ฟังก์ชันปิดโปรแกรม	16
2.13 ฟังก์ชันบันทึกข้อมูลธุรกรรม	17
ภาคผนวก ก Source Code (.py)	18

บทที่ 1

บทนำ

1.1 ชื่อโครงการ

อายุ 20 มีดอยส่วนตัว

1.2 วัตถุประสงค์ของโครงการ

1. เพื่อพัฒนาแอปพลิเคชันสำหรับการจัดการข้อมูลการเทรดคริปโตเคอเรนซีหรือหุ้นอื่นๆ ในการซื้อ การขาย การคำนวณราคาที่เหมาะสมในราคาที่ต่ำที่สุด เพื่อไม่ให้ขาดทุนจากการเทรด
2. เพื่อฝึกทักษะการเขียนโปรแกรมด้วยภาษาไพธอนที่สามารถรองรับการจัดการข้อมูล เช่น การซื้อ การขาย การแสดงผลและ การสร้างไฟล์ เพื่อจัดเก็บข้อมูลการซื้อขาย
3. เป็นการเขียนโปรแกรมเพื่อใช้งานโดยส่วนตัว ที่ชื่นชอบในการเทรดคริปโตเคอเรนซี

1.3 ขอบเขตของโครงการ

แอปพลิเคชันการจัดการข้อมูลการเทรดสำหรับคริปโตเคอเรนซีหรือหุ้น ลักษณะเป็นแอปพลิเคชันสำหรับคอมพิวเตอร์โดยแบ่งออกเป็น ฟังก์ชันซื้อ ฟังก์ชันขาย โดยแบ่งย่อยๆเป็น ขายแบบเปอร์ตเซ็นต์ ขายตามจำนวนเงินFiat และขายเป็นจำนวนหุ้น ฟังก์ชันแสดงผล ฟังก์ชันการลบข้อมูล และฟังก์ชันการบันทึกข้อมูลลงไฟล์

1) ฟังก์ชันซื้อ

- รับค่าค่าธรรมเนียมการเทรด
- รับค่าหน่วยเงินFiat
- รับค่าจำนวนเงินที่ซื้อ
- รับค่าราคาต่อหุ้นหรือคริปโตเคอเรนซี
- แสดงผลสรุปการซื้อ โดยมี ยอดเงินที่ซื้อ ค่าธรรมเนียม ราคาต่อหุ้น จำนวนหุ้นที่ได้รับ รวมหุ้นที่มีอยู่

2) ฟังก์ชันขาย

- แสดงผลยอด/จำนวนที่ขายได้สูงสุด โดยหักค่าธรรมเนียม
- ฟังก์ชันขายแบบจำนวนเปอร์เซ็นต์ โดยรับค่า 0 -100 %
- ฟังก์ชันขายแบบจำนวน Fiat โดยรับค่าตามจำนวนเงินที่ต้องการขาย
- ฟังก์ชันขายแบบจำนวนหุ้น โดยรับค่าตามจำนวนหุ้นที่ต้องการขาย

3) ฟังก์ชันการแสดงผล

- แสดงผลประวัติการซื้อและการขาย
- แสดงผลสรุปโดยรวม

4)ฟังก์ชันลบข้อมูล

- ลบข้อมูลธุรกรรมทั้งหมด

5)ฟังก์ชันบันทึกlogfile

- บันทึกผลประวัติการซื้อและการขาย
- บันทึกผลสรุปโดยรวม
- โดยตรวจสอบจากฟังก์ชันการแสดงผล

คำอธิบายการทำงาน

เริ่มโปรแกรม รับค่าธรรมเนียมของโบรกเกอร์ รับซื้อหุ้น รับสกุลเงิน รับค่าอัตราส่วนของหุ้นต่อสกุลเงิน และรับค่าจำนวนเงินที่ต้องการซื้อ ประมวลผล เพิ่มข้อมูลไปในลิสต์และแสดงผลการซื้อ

การขายมีการรับค่าอัตราส่วนหุ้นต่อสกุลก่อนเพื่อแสดงผลจำนวนที่สามารถขายได้สูงสุดรวมค่าธรรมเนียมแล้ว เพื่อไม่ให้เกิดข้อผิดพลาดหากต้องการจะขายทั้งหมด โดยมีฟังก์ชันให้เลือกทั้งหมดสามฟังก์ชันการขาย โดยฟังก์ชันแรกจะเป็นการขายแบบเปอร์เซ็นต์ มีการป้องกันให้ขายเฉพาะ 0 - 100 % เท่านั้น

ฟังก์ชันที่สองเป็นการขายแบบตามที่กำหนดค่าเงินแบบไม่หักค่าธรรมเนียมโดยจะนำจำนวนที่ใส่ไปคำนวณค่าธรรมเนียมและหักออกทีหลัง จากจำนวนหุ้นที่ถืออยู่ ตัวอย่าง หากใส่ 100 บาท จะได้ 100 บาท แล้วโปรแกรมจะหักค่าธรรมเนียมจากหุ้นที่ถืออยู่แทนจากอัตราส่วนที่รับค่ามาตอนที่ขาย มีการป้องกัน จำนวนที่จะขายไม่ให้เกินจำนวนที่ขายได้สูงสุด

ฟังก์ชันที่สามเป็นการขายแบบจำนวนหุ้น โดยระบบการทำงานจะเป็นเหมือนฟังก์ชันที่สอง เช่น ขาย 1 หุ้น จะหักออกไป 1 หุ้นแต่เงินที่ได้รับจะโดนหักค่าธรรมเนียม มีการป้องกัน จำนวนที่จะขายไม่ให้เกินจำนวนที่ขายได้สูงสุด

การแสดงผลจะแสดงผลธุรกรรมทั้งหมด ทั้งซื้อและขาย และยังสรุปยอด ซื้อ ขาย และค่าธรรมเนียม จำนวนหุ้นที่เหลืออยู่ จากการทำธุรกรรมทั้งหมด

เมนูสุดท้ายเป็นการให้เลือกลีฟังก์ชัน ฟังก์ชันแรกคือซื้อ จะกลับไปรับค่าอัตราต่อค่าเงินจำนวนเงินที่ต้องการซื้อ เพื่อประมวลผลและนำไปเก็บค่าต่อไป ฟังก์ชันล่างข้อมูลเป็นฟังก์ชันที่จะกลับไปถึงการรับค่าธรรมเนียม ซื้อหุ้นและค่าเงิน ฟังก์ชันที่สามปิดโปรแกรม จบการทำงานของโปรแกรมโดยไม่มีการบันทึกใดๆ ฟังก์ชันสุดท้ายเป็นการบันทึกข้อมูลที่แสดงผลเก็บไว้ในไฟล์ เพื่อให้ผู้ใช้ได้เก็บข้อมูลไว้ ดูในอนาคต

2.2 ฟังก์ชันการทำงานเริ่มต้นโปรแกรม

```
Start the Crypto currency calculater
Enter your exchange fees(%) : 0.25
Enter your Fiat : thb
Enter your Token : btc
```

คำอธิบาย

เริ่มต้น โปรแกรมจะรับค่า ค่าธรรมเนียม สกุลเงิน และซื้อหุ้น เพื่อเก็บข้อมูลการเทรด และคำนวณค่าธรรมเนียม

2.3 ฟังก์ชันการซื้อ

```
Enter rate (THB/BTC) : 200000
Enter your spend (THB) : 1000
-----
Spent           :      997.50      THB
Fees            :       -2.5       THB
Rate            :    200000.00    THB/BTC
You received    :     0.004987    BTC
-----
BALANCE         :     0.004987    BTC
-----
```

คำอธิบาย

การซื้อจะนำข้อมูลจากฟังก์ชันเริ่มต้นมาแสดงผลตามอัตราส่วนและสกุลเงิน รับค่าอัตราหุ้นต่อสกุลเงินและเงินที่ต้องการจ่าย เพื่อนำไปคำนวณและแสดงผลสรุปการซื้อ ซึ่งจะประกอบด้วยเงินที่จ่าย ค่าธรรมเนียม อัตราการซื้อต่อสกุลเงิน จำนวนหุ้นที่ได้รับและ จำนวนหุ้นที่มีอยู่

2.4 ฟังก์ชันเมนูขาย

```

-----
SELL
YES  (1)
NO   (2)
SELECT THE MENU : 1
-----

Enter sell rate (THB/BTC) : 2500000
Sell max amount :      0.004975      BTC
Sell max fiat   :      12437.58      THB
-----

|  PERCENT (1)  |  FIAT THB (2)  |  BTC AMOUNT (3)  |
-----

```

คำอธิบาย

หากเลือกขายโปรแกรมจะให้ใส่ค่าอัตราหันต่อสกุลเงินเพื่อนำไปคำนวณจำนวนที่สามารถขายได้สูงสุด เพื่อให้ผู้ใช้เลือกได้ว่าจะขายแบบไหนในเมนูต่อไป หากไม่ขายโปรแกรมจะไปที่ฟังก์ชันการแสดงผลฟังก์ชัน (2.8)

2.5 ฟังก์ชันขายแบบเปอร์เซ็นต์

```

-----
Enter sell rate (THB/BTC) : 2500000
Sell max amount :      0.004975      BTC
Sell max fiat   :      12437.58      THB
-----
|  PERCENT (1)  |  FIAT THB (2)  |  BTC AMOUNT (3)  |
-----
Select sell menu : 1
Enter sell percent (%) : -1
Please enter 0 - 100 %
Enter sell percent (%) : 101
Please enter 0 - 100 %
Enter sell percent (%) : 50
-----
You sold          :      -0.002494      BTC
Fees              :           -15.59      THB
Rate              :      2500000.00      THB/BTC
You Received      :           6218.79      THB
-----
BALANCE           :           0.002494      BTC
-----
SELL MORE ?
YES  (1)
NO   (2)

```

คำอธิบาย

การขายแบบเปอร์เซ็นต์มีการป้องกันการรับค่าที่ต่ำกว่า 0 และมากกว่า 100 เพื่อป้องกันการความผิดพลาดในการคำนวณ และแสดงผลการขาย ประกอบด้วย จำนวนหุ้นที่ขาย ค่าธรรมเนียมการขาย อัตราขายต่อสกุลเงิน จำนวนเงินที่ได้รับ และ จำนวนหุ้นที่เหลืออยู่ และโปรแกรมจะถามว่าขายเพิ่มไหม ถ้าหากขายจะกลับไปฟังก์ชันเมนูขาย(2.4)

2.6 ฟังก์ชันขายแบบจำนวนเงินที่ต้องการรับ

```

Enter sell rate (THB/BTC) : 250000000
Sell max amount : 0.002488 BTC
Sell max fiat : 621878.91 THB
-----
| PERCENT (1) | FIAT THB (2) | BTC AMOUNT (3) |
-----
Select sell menu : 2
Fix THB sell : 10000000000
Your THB not enough
Fix THB sell : -1
Please input more than 0 :
Fix THB sell : 21878.91
-----
You sold : -0.000088 BTC
Fees : -54.70 THB
Rate : 250000000.00 THB/BTC
You Received : 21878.91 THB
-----
BALANCE : 0.002406 BTC
-----
SELL MORE ?
YES (1)
NO (2)
SELECT THE MENU : 

```

คำอธิบาย

การขายแบบจำนวนเงินที่ต้องการรับ จะมีการป้องกันไม่ให้ขายเกิน ค่าที่แสดงผล Sell max fiat หรือคำว่า 0 เพื่อป้องกันการดำเนินงานที่ผิดพลาดของโปรแกรม โดยค่าที่รับมาจะได้เงินตามที่ใส่ค่าลงไป ตามตัวอย่าง โดยค่าธรรมเนียมจะหักจากจำนวนหุ้นที่มีอยู่แทน ในอัตราการขายต่อสกุลเงิน และแสดงผลการขายประกอบด้วย จำนวนหุ้นที่ขาย ค่าธรรมเนียมการขาย อัตราการขายต่อสกุลเงิน จำนวนเงินที่ได้รับ และ จำนวนหุ้นที่เหลืออยู่ และโปรแกรมจะถามว่าขายเพิ่มไหม ถ้าหากขายจะกลับไปฟังก์ชันเมนูขาย(2.4)

2.7 ฟังก์ชันขายแบบจำนวนหุ้น

```

-----
Enter sell rate (THB/BTC) : 2500000
Sell max amount :      0.002400      BTC
Sell max fiat   :      6000.00      THB
-----
|  PERCENT (1)  |  FIAT THB (2)  |  BTC AMOUNT (3)  |
-----
Select sell menu : 3
Enter sell amount : -1
Please input more than 0
Enter sell amount : 10000
Your BTC not enough
Enter sell amount : 0.0004
You sold           :  -0.000400           BTC
Fees               :           -2.50       THB
Rate              : 2500000.00           THB/BTC
You Received      :           997.50       THB
-----
BALANCE           :      0.002006      BTC
-----
SELL MORE ?
YES  (1)
NO   (2)
SELECT THE MENU : 

```

คำอธิบาย

การขายแบบจำนวนหุ้น จะมีการป้องกันไม่ให้ขายเกินค่าที่แสดงผล Sell max amount หรือต่ำกว่า 0 เพื่อป้องกันการดำเนินงานที่ผิดพลาดการคำนวณของโปรแกรม และแสดงผลการขาย ประกอบด้วย จำนวนหุ้นที่ขาย ค่าธรรมเนียมการขาย อัตราขายต่อสกุลเงิน จำนวนเงินที่ได้รับ และ จำนวนหุ้นที่เหลืออยู่ และโปรแกรมจะถามว่าขายเพิ่มไหม ถ้าหากขายจะกลับไปฟังก์ชันเมนูขาย(2.4)

2.8 ฟังก์ชันการแสดงผล

BTC RESULT		
2021-11-17 07:30:52		
FIAT	:	THB
EXCHANGE FEE	:	0.25 %
BUY 1		
SPENT	:	1000.00 THB
BUY RATE	:	200000.00 THB/BTC
FEE	:	-2.50 THB
AMOUNT	:	0.004987 BTC
TOTAL BUY		
SPENT	:	1000.00 THB
FEE	:	-2.50 THB
AMOUNT	:	0.004987 BTC
SELL 1		
RECIEVE	:	6218.79 THB
SELL RATE	:	250000.00 THB/BTC
FEE	:	-15.59 THB
SELL AMOUNT	:	-0.002494 BTC
SELL 2		
RECIEVE	:	21878.91 THB
SELL RATE	:	25000000.00 THB/BTC
FEE	:	-54.70 THB
SELL AMOUNT	:	-0.000088 BTC
SELL 3		
RECIEVE	:	997.50 THB
SELL RATE	:	250000.00 THB/BTC
FEE	:	-2.50 THB
SELL AMOUNT	:	-0.000400 BTC
TOTAL SELL		
RECIEVE	:	29095.20 THB
FEE	:	-72.78 THB
SELL AMOUNT	:	-0.002981 BTC
You total spent	:	1000.00 THB
Total fee	:	-75.28 THB
You amount balance	:	0.002006 BTC
Lower rate sell	:	201002.51 THB/BTC
You recieve form sell	:	29095.20 THB

คำอธิบาย

ฟังก์ชันแสดงผลจะแสดงธุรกรรมที่ผ่านมาทั้งหมดตั้งแต่เริ่มใช้งานโปรแกรมและ สรุปยอดซื้อขายทั้งหมด แสดงเงินที่จ่ายทั้งหมด ค่าธรรมเนียมทั้งหมด จำนวนหุ้นที่เหลืออยู่ อัตราส่วนต่อสกุลเงินที่ต่ำที่สุดที่ไม่ขาดทุนจากเงินที่จ่าย และเงินที่ได้รับจากการขาย

2.9 ฟังก์ชันเมนูสุดท้าย

```

-----
|  BUY MORE (1)  |  RESET (2)  |  QUIT (3)  |
-----
|-----SAVE ALL RESULT TO FILE (4)-----|
-----
Select menu : 

```

คำอธิบาย

เลือกได้ว่าต้องการจะทำอะไรต่อ โดยชื่อเพิ่มจะกลับไปฟังก์ชันชื่อ(2.3) ฟังก์ชันคืนค่าจะกลับไปฟังก์ชันเริ่มต้น(2.2) ฟังก์ชันปิดการทำงานจะปิดการทำงานของโปรแกรม และฟังก์ชันบันทึกลงไฟล์จะไปฟังก์ชันบันทึกลงไฟล์(2.13)

2.10 ฟังก์ชันซื้อเพิ่ม

```

-----
|  BUY MORE (1)  |  RESET (2)  |  QUIT (3)  |
-----
|-----SAVE ALL RESULT TO FILE (4)-----|
-----
Select menu : 1
Enter rate (THB/BTC) : 100
Enter your spend (THB) : 100
-----
Spent           :      99.75      THB
Fees            :      -0.25      THB
Rate            :      100.00     THB/BTC
You received    :      0.997500   BTC
-----
BALANCE         :      0.999506   BTC
-----
SELL
YES  (1)
NO   (2)
SELECT THE MENU : 

```

คำอธิบาย

จะกลับไปฟังก์ชันซื้อ(2.3) โดยข้อมูลก่อนหน้านี้ยังอยู่ครบถ้วน

2.11 ฟังก์ชันลบข้อมูลทั้งหมด

```

-----
|  BUY MORE (1)  |  RESET (2)  |  QUIT (3)  |
-----
|-----SAVE ALL RESULT TO FILE (4)-----|
-----
Select menu : 2
----- RESET COMPLETED -----
Enter your exchange fees(%) : 0.1
Enter your Fait : thb
Enter your Token : ada
Enter rate (THB/ADA) : 100
Enter your spend (THB) : 100
-----
Spent           :      99.90      THB
Fees            :      -0.1      THB
Rate            :      100.00     THB/ADA
You received    :      0.999000   ADA
-----
BALANCE         :      0.999000   ADA
-----
SELL
YES  (1)
NO   (2)
SELECT THE MENU : 

```

คำอธิบาย

ลบประวัติธุรกรรมทั้งหมดและกลับไปฟังก์ชันเริ่มต้น(2.2)

2.12 ฟังก์ชันปิดโปรแกรม

```
-----  
|  BUY MORE (1)  |  RESET (2)  |  QUIT (3)  |  
-----  
|-----SAVE ALL RESULT TO FILE (4)-----|  
-----  
Select menu : 3  
----- EXIT PROGRAM -----
```

คำอธิบาย

ปิดการทำงานของโปรแกรมโดยไม่บันทึกค่าใดๆ

2.13 ฟังก์ชันบันทึกข้อมูลธุรกรรม

1	
2	
3	ADA RESULT
4	
5	2021-11-17 08:31:20
6	FIAT : THB
7	EXCHANGE FEE : 0.10 %
8	
9	BUY 1
10	SPENT : 100.00 THB
11	BUY RATE : 100.00 THB/ADA
12	FEE : -0.10 THB
13	AMOUNT : 0.999000 ADA
14	
15	TOTAL BUY
16	SPENT : 100.00 THB
17	FEE : -0.10 THB
18	AMOUNT : 0.999000 ADA
19	
20	
21	TOTAL SELL
22	RECIEVE : 0.00 THB
23	FEE : 0.00 THB
24	SELL AMOUNT : 0.000000 ADA
25	
26	You total spent : 100.00 THB
27	Total fee : -0.10 THB
28	You amount balance : 0.999000 ADA
29	Lower rate sell : 100.20 THB/ADA
30	You recieve form sell : 0.00 THB
31	
32	

คำอธิบาย

บันทึกข้อมูลธุรกรรมลงไฟล์ โดยค่าที่จะบันทึกจะสามารถตรวจสอบได้จากฟังก์ชันแสดงผล(2.8) ก่อนที่จะบันทึก และปิดการทำงานโปรแกรม

ภาคผนวก

```

import datetime;
ct = datetime.datetime.now()
ct = str(ct)
time = (ct[0:-7])

def buy (fee,cur,coin) :
    rate = float(input("Enter rate (%s/%s) : "%(cur,coin)))
    while rate <= 0 :
        print("Please enter rate more than 0 %s/%s "%(cur,coin))
        rate = float(input("Enter rate (%s/%s) : "%(cur,coin)))
    spend = float(input("Enter your spend (%s) : "%(cur)))
    while spend <= 0 :
        print("Please enter spend more than 0 %s"%(cur))
        spend = float(input("Enter your spend (%s) : "%(cur)))
    spendlist.append(spend)
    buyratelist.append(rate)
    totalfee = spend * fee
    totalspend = spend - totalfee
    amt = totalspend/rate
    amtlist.append(amt)
    feebuylist.append(totalfee)
    print("-"*51)
    print("Spent                : %10.2f \t %s"%(totalspend,cur))
    print("Fees                  : %10s \t %s"%(-totalfee,cur))
    print("Rate                   : %10.2f \t %s/%s"%(rate,cur,coin))
    print("You received           : %10f \t %s"%(amt,coin))
    print("-"*51)
    print("BALANCE                : %10f \t %s"%(sum(amtlist)-sum(selllist),coin))
    print("-"*51)
    return (spend,rate)

def funper (amtlist,sellrate,fee) :
    sellper = float(input("Enter sell percent (%) : "))
    while sellper < 0 or sellper > 100 :
        print("Please enter 0 - 100 %")
        sellper = float(input("Enter sell percent (%) : "))
    per = sellper/100
    sell = (sum(amtlist)-sum(selllist)) * per
    totalsell = sell * sellrate
    totalfee = totalsell * fee
    rec = totalsell - totalfee
    reclist.append(rec)
    selllist.append(sell)
    feeselllist.append(totalfee)
    print("-"*51)
    print("You sold                : %10f \t %s"%(-sell,coin))
    print("Fees                   : %10.2f \t %s"%(-totalfee,cur))
    print("Rate                   : %10.2f \t %s/%s"%(sellrate,cur,coin))
    print("You Received           : %10.2f \t %s"%(rec,cur))
    print("-"*51)
    print("BALANCE                : %10f \t %s"%(sum(amtlist)-sum(selllist),coin))
    print("-"*51)

```

```

def funfiat(maxfiatsell,sellrate,fee) :
    sellfix = float(input("Fix %s sell :"%(cur)))
    while maxfiatsell < sellfix :
        print("Your %s not enough"%(cur))
        sellfix = float(input("Fix %s sell :"%(cur)))
    while sellfix < 0 :
        print("Please input more than 0 : ")
        sellfix = float(input("Fix %s sell :"%(cur)))
    sell = sellfix/sellrate
    totalfee = sellfix * fee
    totalsell = sell + (sell*fee)
    reclist.append(sellfix)
    selllist.append(totalsell)
    feeselllist.append(totalfee)
    print("-"*51)
    print("You sold           : %10f \t %s"%(-totalsell,coin))
    print("Fees                 : %10.2f \t %s"%(-totalfee,cur))
    print("Rate                  : %10.2f \t %s/%s"%(sellrate,cur,coin))
    print("You Received          : %10.2f \t %s"%(sellfix,cur))
    print("-"*51)
    print("BALANCE               : %10f \t %s"%(sum(amtlist)-sum(selllist),coin))
    print("-"*51)

```

```

def funamt (maxamtsell,sellrate,fee):
    sellamt = float(input("Enter sell amount : "))
    while sellamt < 0 :
        print("Please input more than 0")
        sellamt = float(input("Enter sell amount : "))
    while maxamtsell < sellamt :
        print("Your %s not enough"%(coin))
        sellamt = float(input("Enter sell amount : "))
    sell = sellamt * sellrate + (sellamt * sellrate * fee)
    totalfee = sell*fee
    totalsell = sell - totalfee
    totalamt = sellamt+(sellamt*fee)
    selllist.append(totalamt)
    feeselllist.append(totalfee)
    reclist.append(totalsell)
    print("You sold           : %10f \t %s"%(-sellamt,coin))
    print("Fees                 : %10.2f \t %s"%(-totalfee,cur))
    print("Rate                  : %10.2f \t %s/%s"%(sellrate,cur,coin))
    print("You Received          : %10.2f \t %s"%(totalsell,cur))
    print("-"*51)
    print("BALANCE               : %10f \t %s"%(sum(amtlist)-sum(selllist),coin))
    print("-"*51)

```

```

def result (coin,time,cur,feebuylist,exfee,spendlist,buyratelist,amtlist,reclist,sellratelist,feeselllist,selllist) :
    print("-"*51)
    print(" %s RESULT ".center(51)% (coin))
    print("-"*51)
    print(time)
    print("FIAT                : %10s"%(cur))
    print("EXCHANGE FEE           : %10.2f %s"%(exfee,"%"))
    print("-"*51)
    for i in range(len(feebuylist)) :
        print("BUY %d".center(51)% (i+1))
        print("SPENT                : %10.2f \t %s"%(spendlist[i],cur))
        print("BUY RATE              : %10.2f \t %s/%s"%(buyratelist[i],cur,coin))
        print("FEE                   : %10.2f \t %s"%(-(feebuylist[i]),cur))
        print("AMOUNT                : %10f \t %s"%(amtlist[i],coin))
        print("-"*51)
    print("TOTAL BUY".center(51))
    print("SPENT                : %10.2f \t %s"%(sum(spendlist),cur))
    print("FEE                   : %10.2f \t %s"%(-(sum(feebuylist)),cur))
    print("AMOUNT                : %10f \t %s"%(sum(amtlist),coin))
    print("-"*51)
    for i in range(len(feeselllist)) :
        print("SELL %d".center(51)% (i+1))
        print("RECIEVE              : %10.2f \t %s"%(reclist[i],cur))
        print("SELL RATE             : %10.2f \t %s/%s"%(sellratelist[i],cur,coin))
        print("FEE                   : %10.2f \t %s"%(-(feeselllist[i]),cur))
        print("SELL AMOUNT           : %10f \t %s"%(-(selllist[i],coin))
        print("-"*51)
    print("TOTAL SELL".center(51))
    print("RECIEVE              : %10.2f \t %s"%(sum(reclist),cur))
    print("FEE                   : %10.2f \t %s"%(-(sum(feeselllist)),cur))
    print("SELL AMOUNT           : %10f \t %s"%(-(sum(selllist),coin))
    print("-"*51)
    print("You total spent      : %10.2f \t %s"%(sum(spendlist),cur))
    print("Total fee            : %10.2f \t %s"%(-(sum(feebuylist)+(sum(feeselllist))),cur))
    print("You amount balance   : %10f \t %s"%(sum(amtlist)-sum(selllist),coin))
    print("Lower rate sell      : %10.2f \t %s/%s"
          "%((float((sum(spendlist))/(sum(amtlist)))+(((sum(spendlist))/(sum(amtlist))*fee))),cur,coin))
    print("You recieve form sell : %10.2f \t %s"%(sum(reclist),cur))
    print("-"*51)

```

```

def clear (spendlist,feebuylist,feeselllist,amtlist,selllist,reclist,buyratelist,sellratelist) :
    spendlist.clear()
    feebuylist.clear()
    feeselllist.clear()
    amtlist.clear()
    selllist.clear()
    reclist.clear()
    buyratelist.clear()
    sellratelist.clear()

```

```

def save (coin,time,cur,feebuylist,exfee,spendlist,buyratelist,amtlist,reclist,sellratelist,feeselllist,selllist) :
    with open("CRYPTO.txt","a+") as file :
        file.write("\n")
        file.write("-"*51)
        file.write("\n")
        file.write("%s RESULT".center(51)% (coin))
        file.write("\n")
        file.write("-"*51)
        file.write("\nFIAT                : %10s\n" % (cur))
        file.write("EXCHANGE FEE          : %10.2f %s\n" % (exfee,"%"))
        file.write("%s\n" % (time))
        file.write("-"*51)
        file.write("\n")
        for i in range(len(feebuylist)) :
            file.write("BUY %d".center(51)% (i+1))
            file.write("\nSPENT                : %10.2f \t %s\n" % (spendlist[i],cur))
            file.write("BUY RATE                : %10.2f \t %s/%s\n" % (buyratelist[i],cur,coin))
            file.write("FEE                    : %10.2f \t %s\n" % (-feebuylist[i],cur))
            file.write("AMOUNT                : %10f \t %s\n" % (amtlist[i],coin))
            file.write("-"*51)
        file.write("\n")
        file.write("TOTAL BUY".center(51))
        file.write("\nSPENT                : %10.2f \t %s\n" % (sum(spendlist),cur))
        file.write("FEE                    : %10.2f \t %s\n" % (-sum(feebuylist),cur))
        file.write("AMOUNT                : %10f \t %s\n" % (sum(amtlist),coin))
        file.write("-"*51)
        file.write("\n")
        for i in range(len(feeselllist)) :
            file.write("SELL %d".center(51)% (i+1))
            file.write("\nRECEIVE                : %10.2f \t %s\n" % (reclist[i],cur))
            file.write("SELL RATE                : %10.2f \t %s/%s\n" % (sellratelist[i],cur,coin))
            file.write("FEE                    : %10.2f \t %s\n" % (-feeselllist[i],cur))
            file.write("SELL AMOUNT            : %10f \t %s\n" % (selllist[i],coin))
            file.write("-"*51)
        file.write("\n")
        file.write("TOTAL SELL".center(51))
        file.write("\nRECEIVE                : %10.2f \t %s\n" % (sum(reclist),cur))
        file.write("FEE                    : %10.2f \t %s\n" % (-sum(feeselllist),cur))
        file.write("SELL AMOUNT            : %10f \t %s\n" % (sum(selllist),coin))
        file.write("-"*51)
        file.write("\nYou total spent        : %10.2f \t %s\n" % (sum(spendlist),cur))
        file.write("Total fee              : %10.2f \t %s\n" % (-((sum(feebuylist)+(sum(feeselllist)))),cur))
        file.write("You amount balance    : %10f \t %s\n" % (sum(amtlist)-sum(selllist),coin))
        file.write("Lower rate sell       : %10.2f \t %s/%s\n" % ((float((sum(spendlist))/(sum(amtlist))+((sum(spendlist))/(sum(amtlist))*fee))),cur,coin))
        file.write("You recieve form sell : %10.2f \t %s\n" % (sum(reclist),cur))
        file.write("-"*51)
        file.write("\n")
    print("----- SAVE TO FILE COMPLETE -----")

```

```

spendlist = []
feebuylist = []
feeselllist = []
amtlist = []
selllist = []
reclist = []
buyratelist = []
sellratelist = []
maxamtsell,maxfiatsell=(0,0)

print("Start the Crypto currency calulater")
exfee = float(input("Enter your exchange fees(%) : "))
while exfee < 0 or exfee > 100:
    print("Please enter 0 - 100 % ")
    exfee = float(input("Enter your exchange fees(%) : "))
fee = exfee/100
cur = input("Enter your Fait : ").upper()
coin = input("Enter your Token : ").upper()
start = "S"
while start != "N":
    buy(fee,cur,coin)
    print("SELL ")
    print("YES  (1)")
    print("NO   (2)")
    sell = input("SELECT THE MENU : ").upper()
    print("-"*51)
    while sell != "1" and sell != "2" :
        sell = input("Please select 1 or 2 ").upper()
    while sell == "1" :
        sellrate = float(input("Enter sell rate (%s/%s) : "%(cur,coin)))
        while sellrate <= 0 :
            sellrate = float(input("Enter sell rate more than 0 (%s/%s) : "%(cur,coin)))
        sellratelist.append(sellrate)
        maxamtsell = (sum(amtlist)-sum(selllist)) - ((sum(amtlist)-sum(selllist)) * fee)
        maxfiatsell = ((sum(amtlist)-sum(selllist))*sellrate) - ((sum(amtlist)-sum(selllist)) * sellrate * fee)
        print("Sell max amount : %10f \t %s"%(maxamtsell,coin))
        print("Sell max fiat : %10.2f \t %s"%(maxfiatsell,cur))
        print("-"*51)
        print("| PERCENT (1) | FIAT %s (2) | %s AMOUNT (3) |"%(cur,coin))
        print("-"*51)
        menu = input("Select sell menu : ").upper()
        while menu != "1" and menu != "2" and menu != "3" :
            menu = input("Please select (1)(2)(3) : ").upper()
        if menu == "1" :
            funper (amtlist,sellrate,fee)
        elif menu == "2" :
            funfiat(maxfiatsell,sellrate,fee)
        elif menu == "3" :
            funamt(maxamtsell,sellrate,fee)
        print("-"*51)
        print("SELL MORE ? ")
        print("YES  (1)")
        print("NO   (2)")
        sell = input("SELECT THE MENU : ").upper()

```

```

print("-"*51)
while sell != "1" and sell != "2" :
    sell = input("Please select (1)(2) : ").upper()
result (coin,time,cur,feebuylist,exfee,spendlist,buyratelist,amtlist,reclist,sellratelist,feeselllist,selllist,)
print()
print("-"*51)
print("| BUY MORE (1) | RESET (2) | QUIT (3) |")
print("-"*51)
print("|-----SAVE ALL RESULT TO FILE (4)-----|")
print("-"*51)
endmenu = input("Select menu : ").upper()
while endmenu != "1" and endmenu != "2" and endmenu != "3" and endmenu != "4" :
    endmenu = input("Please select (1)(2)(3)(4) : ")
if endmenu == "1" :
    start != "N"
elif endmenu == "2" :
    clear(spendlist,feebuylist,feeselllist,amtlist,selllist,reclist,buyratelist,sellratelist)
    print("----- RESET COMPLETED -----")
    exfee = float(input("Enter your exchange fees(%) : "))
    while exfee < 0 or exfee > 100:
        print("Please enter 0 - 100 % ")
        exfee = float(input("Enter your exchange fees(%) : "))
    fee = exfee/100
    cur = input("Enter your Fait : ").upper()
    coin = input("Enter your Token : ").upper()
elif endmenu == "3" :
    start = "N"
elif endmenu == "4" :
    save (coin,time,cur,feebuylist,exfee,spendlist,buyratelist,amtlist,reclist,sellratelist,feeselllist,selllist)
    start = "N"
print("----- EXIT PROGRAM -----")

```