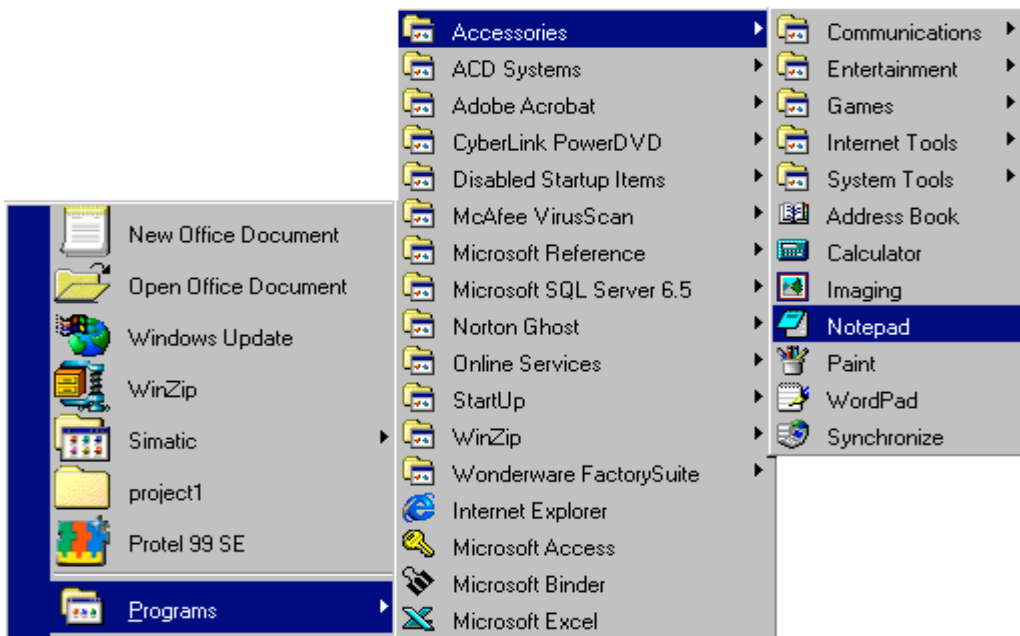


การติดต่อกับฐานข้อมูลชนิด text file

การใช้ฐานข้อมูลที่เป็น text file ถือเป็นที่นิยมมาตั้งแต่อดีตกาลจนถึงปัจจุบันเห็นได้จากโปรแกรมสำเร็จรูปต่างจนไปถึงระบบปฏิบัติการ windows ก็ยังใช้ text file เป็นฐานข้อมูลในการสนับสนุนการทำงานของโปรแกรม ที่สำคัญการเข้าถึงฐานข้อมูลที่เป็น text file ทำได้ง่ายและสามารถใช้คำสั่ง Type ของ Dos ดูข้อมูลได้ ผู้เขียนขอให้ท่านทำความเข้าใจกับเนื้อหาสาระในบทนี้ให้มาก ๆ เพราะมันจะเป็นตัวช่วยให้งานของท่านในอนาคตสำเร็จลุล่วงลงได้อย่างง่ายดาย

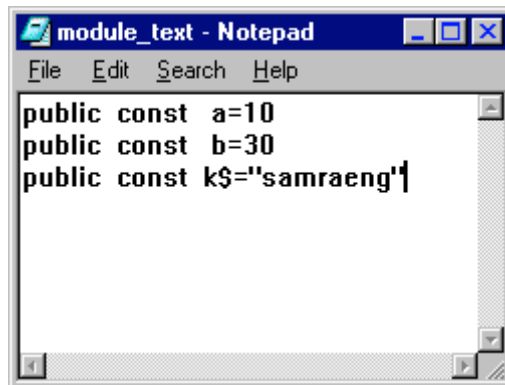
5.1 วิธีการใช้ text file สร้างตัวแปรแบบค่าคงที่

- ใช้ note pad ในการพิมพ์ข้อความ



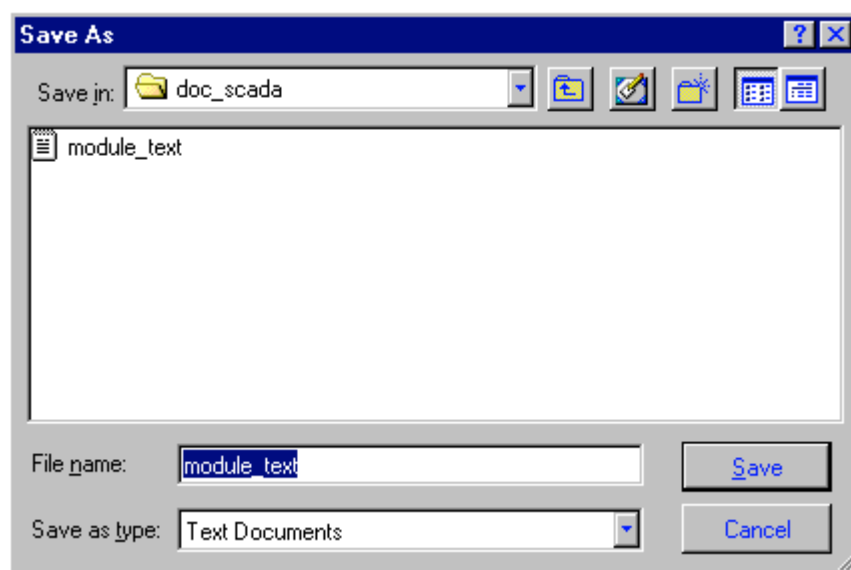
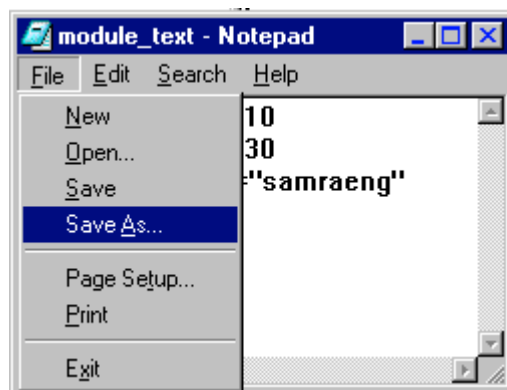
รูปที่ 5.1 การเปิดโปรแกรม note pad

- พิมพ์ข้อความตามรูป 5.2



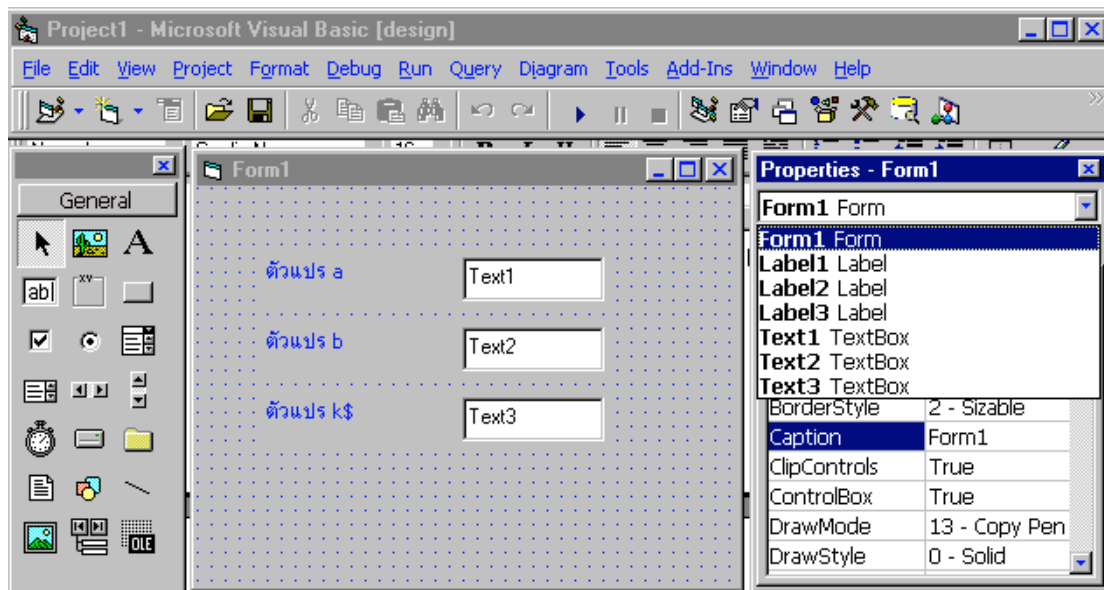
รูป 5.2 ประกาศตัวแปรตามไวยากรณ์ visual basic

- save file on hadr disk



รูป 5.3 บันทึก text file

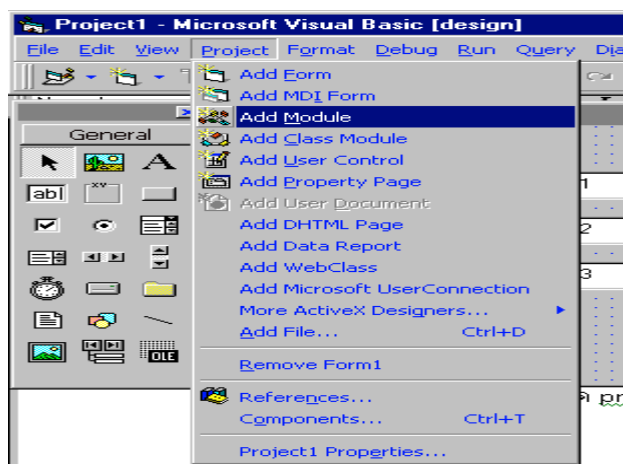
5.2 การ add text file มาที่ Module ใน visual basic



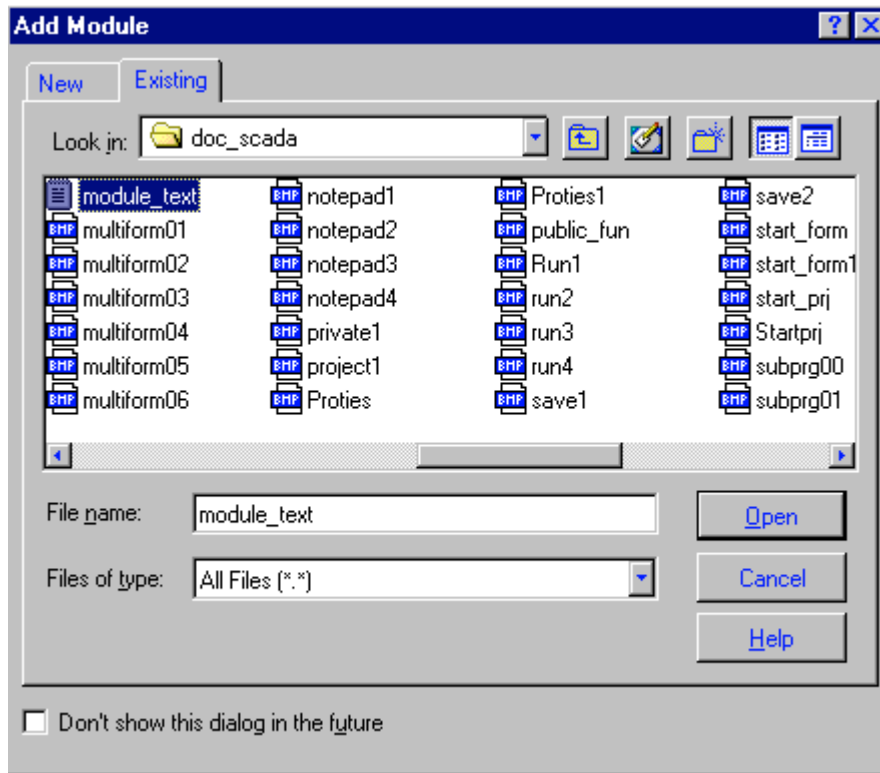
รูป 5.4 วาดคอนโทรล label และ text box ลงบน form

tool	property	defind
Label1	caption	ตัวแปร a
Label2	caption	ตัวแปร b
Label3	caption	ตัวแปร k\$

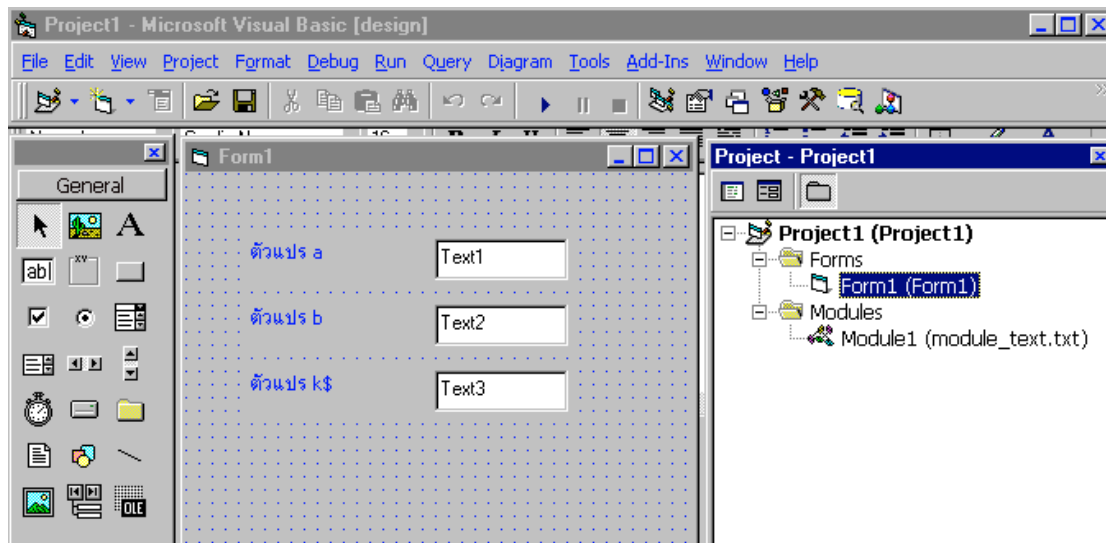
ตาราง 5.1 การกำหนด property ให้กับวัตถุผ่านทาง property windows



รูป 5.5 การใช้เมนูเพื่อ add module



รูป 5.5 เปิด text file ที่ใช้ในการประกาศตัวแปร



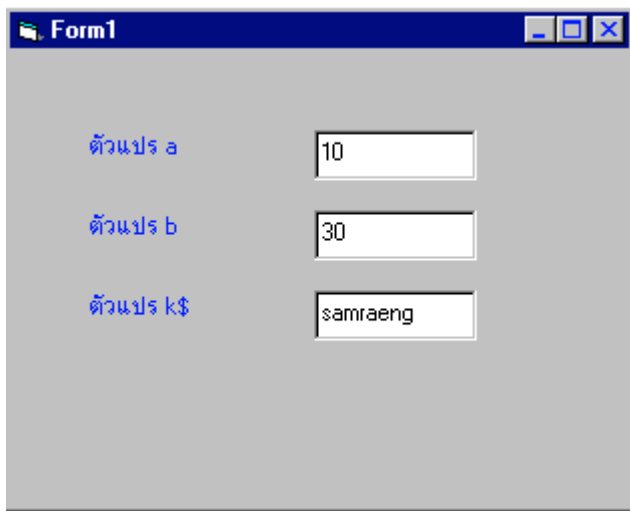
รูป 5.6 ที่ module1 เป็น text file ที่เก็บค่าตัวแปรแบบ const

```

Project1 - Form1 (Code)
Form Load
Private Sub Form_Load()
    Text1 = a
    Text2 = b
    Text3 = k$
End Sub

```

รูป 5.7 ทดสอบเขียนโปรแกรมติดต่อกับตัวแปรใน module1



รูป 5.8 run program ตัวแปร a,b และ k\$ จะแสดงที่ text1,text2 และ text3 ตามลำดับ

5.3 การเปิดและปิด File

ก่อนที่จะทำการบันทึกหรือเรียกดูข้อมูลเราจะต้องทำการเปิดข้อมูลก่อนเสมอ file ที่ ถูกเปิด จะถูกเก็บไว้ในหน่วยความจำ ซึ่งถูกแบ่งออกเป็นพื้นที่ทำให้เราสามารถเปิด file ได้หลาย ๆ file ในเวลาเดียวกันได้ รูปแบบคำสั่งที่ใช้ในการเปิด file มีดังนี้

Open <File name> for <mode> [access access] [lock] As [#] File number_
[len=reclength]

โดยที่

file name เป็นชื่อ file ตาม directory ที่ต้องการเปิด เช่น "c:\sam1\Module_text.txt"

เป็นต้น

Mode ใช้กำหนดรูปแบบในการเปิด file

- Append ใช้ในกรณีบันทึกต่อท้าย file
- ใช้ในกรณีที่ต้องการเปิด binary file
- input ใช้ในกรณีเปิด file เพื่ออ่านข้อมูล
- output ใช้ในกรณี เปิด file เพื่อบันทึกข้อมูล
- (ค่า default) ใช้ในกรณีที่ต้องการเปิด random access file

Access จะระบุหรือไม่ก็ได้ใช้กำหนดการทำงานที่สามารถกระทำกับ file นั้นได้

- read ใช้กำหนดให้ file ที่เปิดอ่านข้อมูลได้อย่างเดียว
- write ใช้กำหนดให้ file ที่เปิดบันทึกข้อมูลได้อย่างเดียว
- read write ใช้กำหนดให้ file ที่เปิดสามารถอ่านและเขียนข้อมูลได้

Lock จะระบุหรือไม่ก็ได้ ใช้กำหนดให้ Lock File ก่อนดำเนินการใด ๆ กับ file ในกรณีเปิด file แบบ Multi-user

- share กำหนดให้ file นั้นสามารถเปิดได้โดย User หลาย ๆ คน
- Lock Read กำหนดให้ file นั้นใช้ได้เพียง User เดียว และใช้ในการอ่านเท่านั้น
- Lock Write กำหนดให้ File นั้นใช้ได้เพียง User เดียว และใช้ในการบันทึกเท่านั้น
- Lock Read Write กำหนดให้ใช้ file ได้ เพียง User เดียวโดยใช้ได้ทั้งอ่านและเขียน

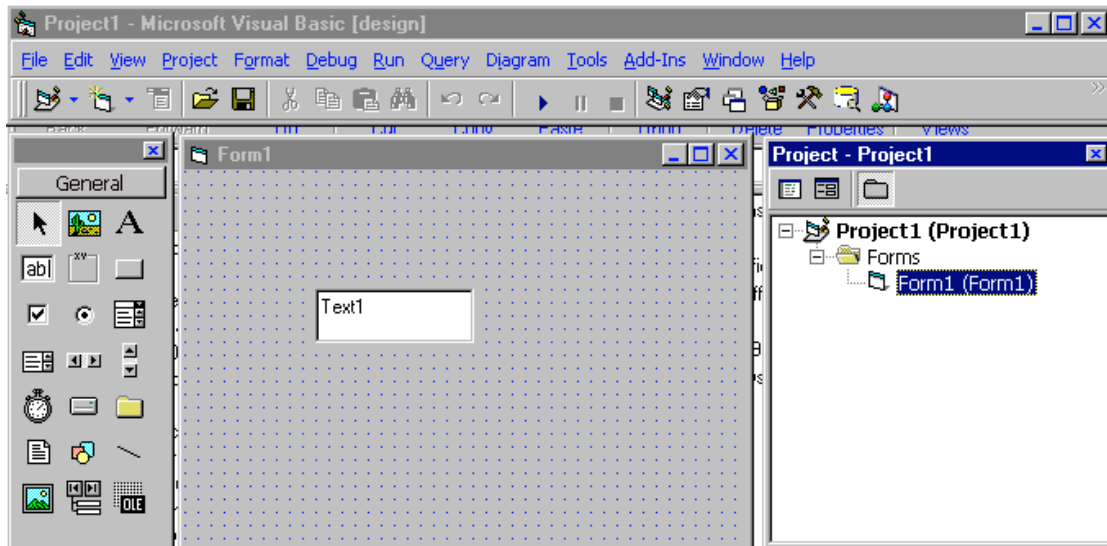
Filenumber เป็นหมายเลขพื้นที่ในหน่วยความจำที่จะใช้เก็บ File ที่เปิดซึ่งมีค่าอยู่ระหว่าง 1-511 ในการกำหนด จะระบุพื้นที่หลังเครื่องหมาย “#”

Len จะระบุหรือไม่ก็ได้โดยมี 2 ความหมายดังนี้

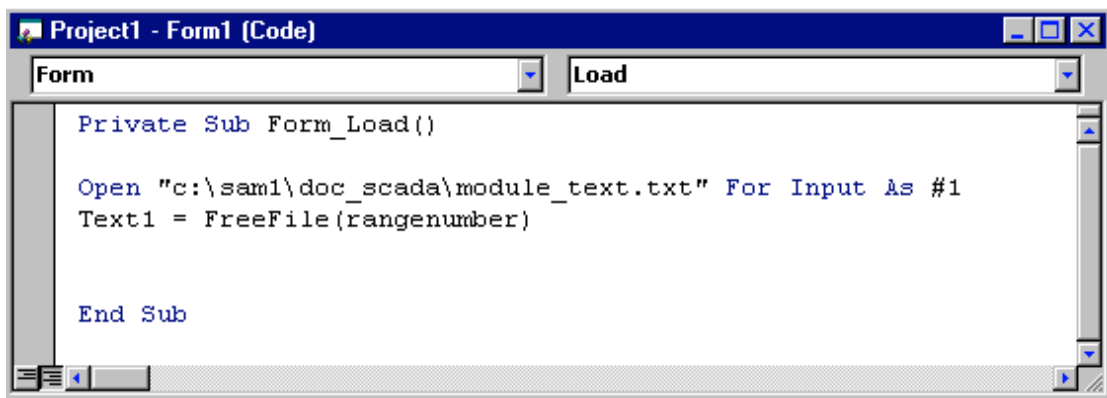
- ขนาดของ record ในกรณีเปิด file แบบ random
- ขนาดของ file ในกรณีเปิด file แบบ sequential

file ที่ถูกเปิดจะถูกเก็บไว้ในหน่วยความจำซึ่งถูกแบ่งออกเป็นพื้นที่จำนวน 511 พื้นที่ พื้นที่ที่ใช้เก็บ file จะไม่สามารถนำมาใช้ได้อีกจนกว่าพื้นที่นั้นจะถูกยกเลิกดังนั้นในกรณีที่ต้องการเปิด file เพิ่ม จะต้องทราบว่าพื้นที่ใดว่างบ้างโดยใช้ฟังก์ชัน FreeFile ตรวจสอบ

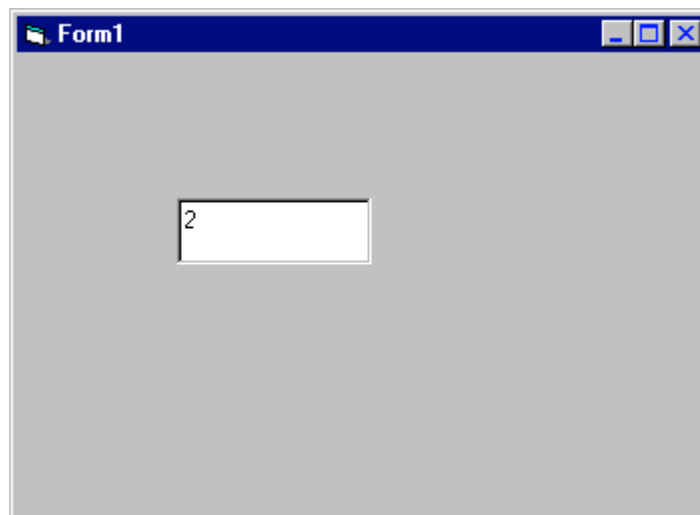
ตัวอย่าง 5.1



รูป 5.9 สร้าง project ตามรูป



รูป 5.10 เขียนโปรแกรม ที่ sub form_load



รูป 5.11 text1 แสดงพื้นที่ว่างในหน่วยความจำ

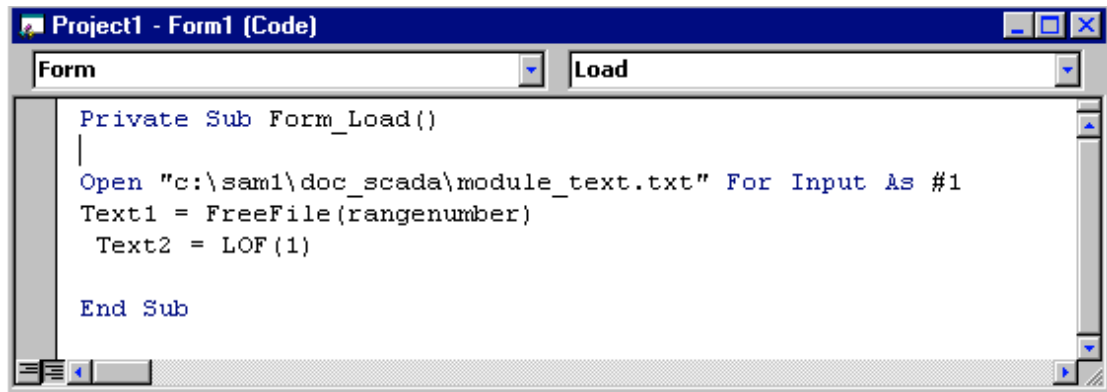
ในกรณีที่ต้องการยกเลิกใช้งาน file ใด เราสามารถปิด file นั้นแล้วคืนพื้นที่นั้นให้กับหน่วยความจำ ด้วยคำสั่ง Close [#filenumber] เช่นเมื่อต้องการปิด file หมายเลข 1 จะใช้ Close #1 เป็นต้น

5.4 การดูขนาดของ File

เมื่ออยากทราบขนาดของ File ที่ทำการเปิดขึ้นมาใช้งาน ทำได้โดยการใช้คำสั่ง LOF ซึ่งมีรูปแบบดังนี้

```
LOF(#FILENUMBER)
```

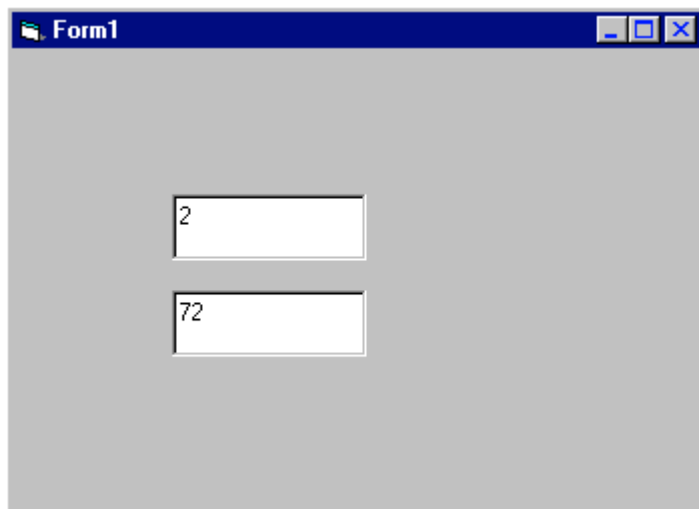
จาก project ในตัวอย่าง 5.1 ให้เพิ่ม text2 เข้ามาในโปรเจกต์แล้วเขียนโปรแกรมเพิ่มเติมดังรูป



```
Project1 - Form1 (Code)
Form Load
Private Sub Form_Load()
|
Open "c:\sam1\doc_scada\module_text.txt" For Input As #1
Text1 = FreeFile(rangenumber)
Text2 = LOF(1)

End Sub
```

รูป 5.12 การใช้คำสั่ง LOF(#Filenumber) เพื่อดูขนาดของ file



Form1

2

72

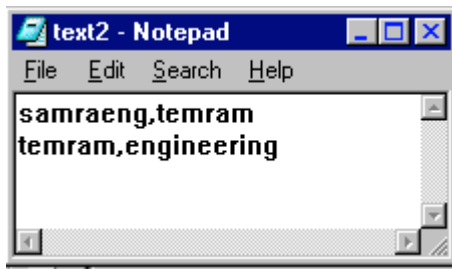
รูป 5.13 text2 บอกรายละเอียดของ file #1

5.5 การอ่านข้อมูลจาก Sequential File

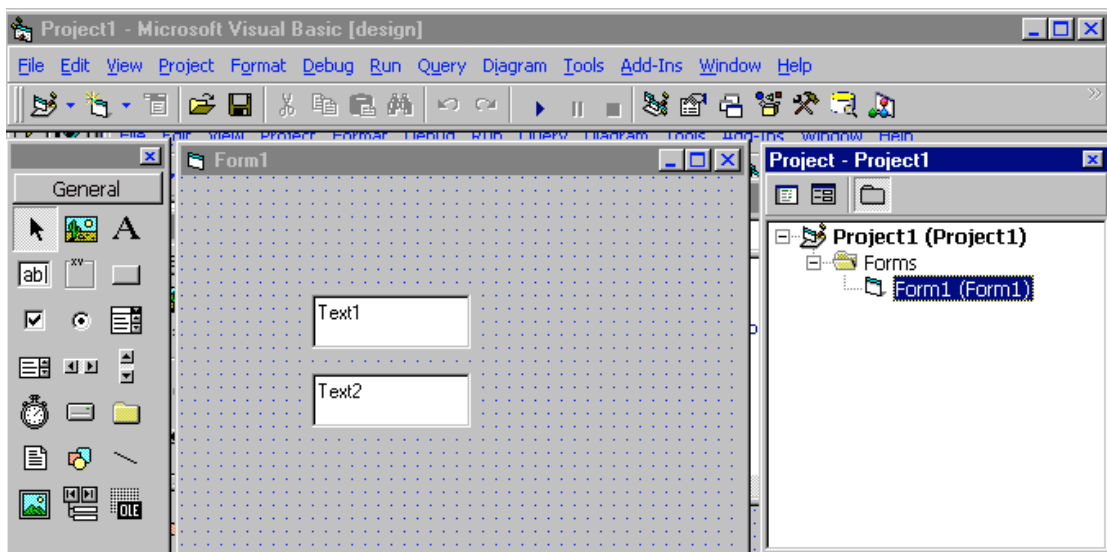
ขั้นตอนการอ่านข้อมูลจาก Sequential File

- Line Input #filenumber เป็นคำสั่งที่ใช้อ่านข้อมูล 1 บรรทัด
- input #filenumber เป็นคำสั่งที่ใช้อ่านข้อมูลที่ถูกแบ่งด้วย เครื่องหมาย “,”

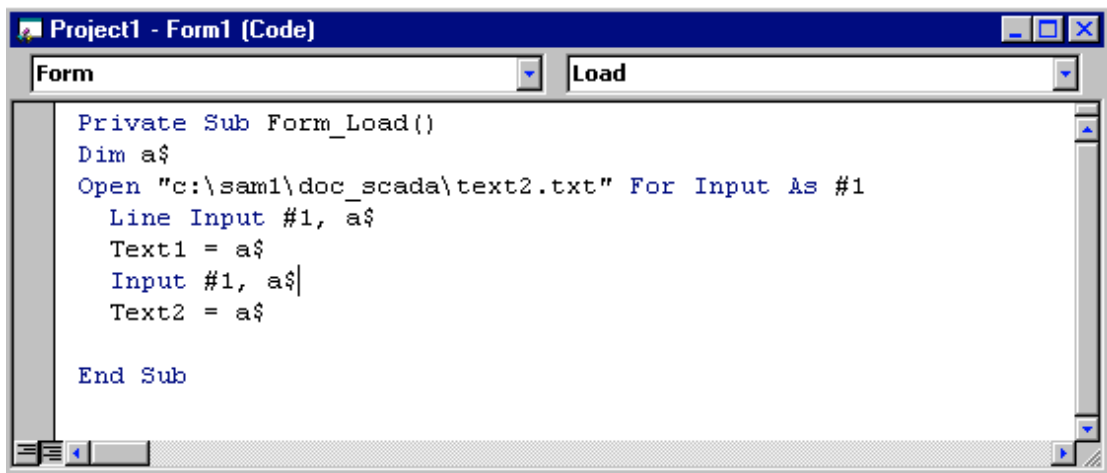
ตัวอย่าง 5.2



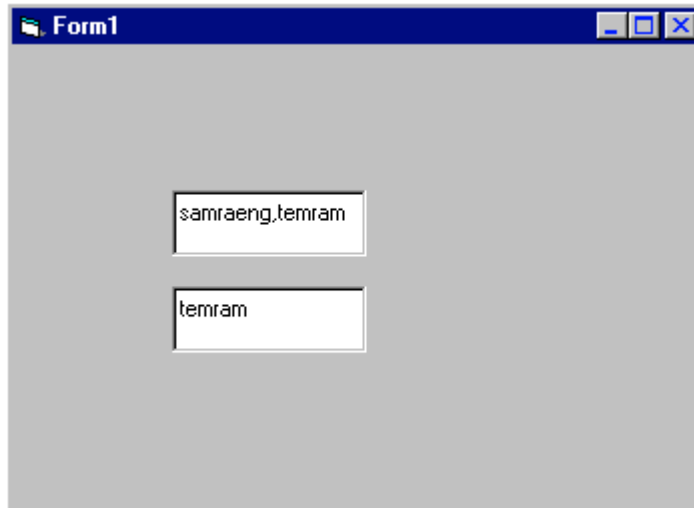
รูป 5.14 สร้าง textfile โดยใช้ notepad



รูป 5.14 สร้าง project ประกอบด้วย 2 textbox



รูป 5.15 การใช้คำสั่ง input และ line input

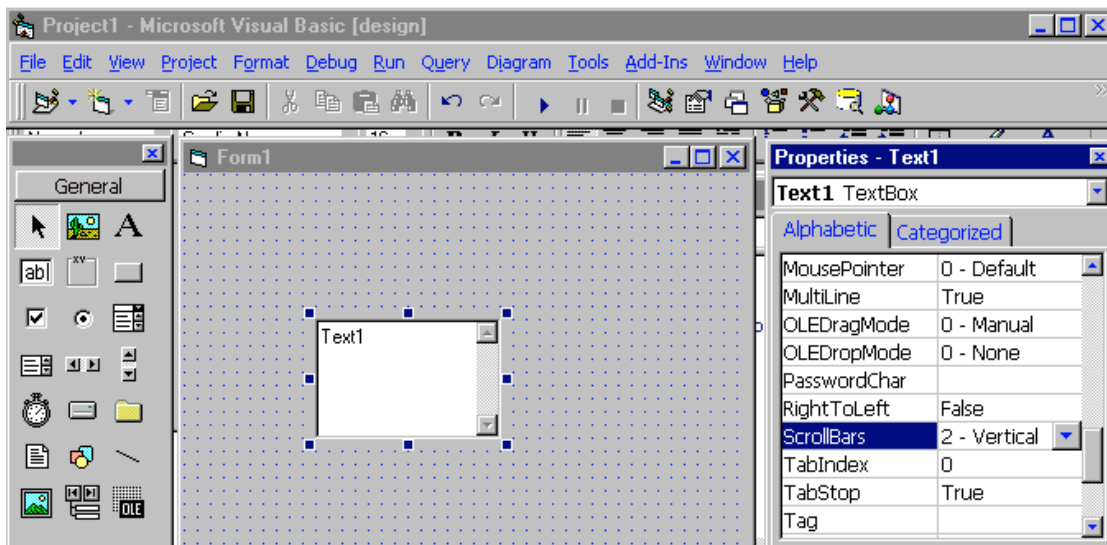


รูป 5.16 แสดงความแตกต่างระหว่าง input และ line input

5.6 EOF(N)

ย่อมาจาก End Of File คำสั่งสำหรับตรวจสอบการสิ้นสุดฐานข้อมูลเมื่อ N เป็นหมายเลข file

ตัวอย่าง 5.3

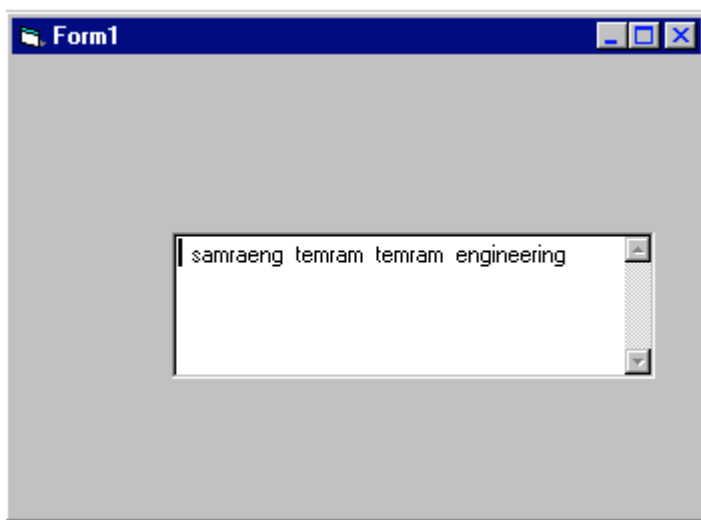


รูป 5.17

object	property	defind
Text1	mutiline	true
	scrollbar	vertical

```
Private Sub Form_Load()  
Dim a$  
Open "c:\sam1\doc_scada\text2.txt" For Input As #1  
Do While Not EOF(1)  
  
    Input #1, a$  
    Text1 = Text1 + " " + " " + a$  
Loop  
  
End Sub
```

รูป 5.18



รูป 5.19 พิจารณาเทียบกับรูป 5.14

5.6 การบันทึกข้อมูลลงใน text file

การบันทึกข้อมูลชนิด text file ลงใน hard disk ทำได้โดยการเปิด file แล้วตามด้วย Mode จาก input เป็น

- output เมื่อต้องการบันทึกทับข้อมูลเดิม
- Append เมื่อต้องการบันทึกต่อท้ายข้อมูลเดิม

ตัวอย่าง 5.4 แสดงการเขียนโปรแกรมเพื่อบันทึกข้อมูลชนิด text file

```

Project1 - Form1 (Code)
Form Load
Private Sub Form_Load()
    Open "c:\sam1\doc_scada\test_write.txt" For Output As #1

    Print #1, " hello Mr samraeng"
    Print #1, " how are you to day"
    Close #1
    |
End Sub

```

รูป 5.20 แสดงการเขียน program เพื่อพิมพ์ข้อความ 2 บรรทัดลงใน file ชื่อ "test_write.txt"

หลังจาก Run program ลองใช้ program note pad เปิดดูข้อความใน file ดังกล่าว

```

Project1 - Form1 (Code)
Form Load
Private Sub Form_Load()
    Open "c:\sam1\doc_scada\test_write.txt" For Append As #1

    Print #1, "======"
    Print #1, " this is last updat data"
    Close #1
    |
End Sub

```

รูป 5.21 แสดงการเขียนโปรแกรมเพื่อบันทึกข้อมูลต่อจากข้อมูลเดิม

หลังจาก Run program ลองใช้ program note pad เปิดดูข้อความใน file ดังกล่าวอีกครั้ง

