



קבצים בשפת Visual Basic

ניר אדר

מסמך זה הורד מהאתר <http://underwar.livedns.co.il>

אין להפיץ מסמך זה במדיה כלשהי, ללא אישור מפורש מאת המחבר.

מחבר המסמך איננו אחראי לכל נזק, ישיר או עקיף, שיגרם עקב השימוש במידע המופיע במסמך, וכן לנכונות התוכן של הנושאים המופיעים במסמך. עם זאת, המחבר עשה את מירב המאמצים כדי לספק את המידע המדויק והמלא ביותר.

כל הזכויות שמורות לניר אדר

Nir Adar

Email: underwar@hotmail.com

Home Page: <http://underwar.livedns.co.il>

אנא שלחו תיקונים והערות אל המחבר.

קבצים ב-Visual Basic

כאשר אנו כותבים תוכנית ויוצרים בה משתנים, המשתנים נשמרים בזיכרון המהיר של המחשב (RAM). כאשר התוכנית תסתיים, כל המשתנים עם תכולתם ייעלמו, וכאשר נריץ שוב את התוכנית, היא תיצור משתנים חדשים משלה.

לעתים נרצה דרך לשמור מידע - כך שנוכל להחזירו גם בריצות הבאות של התוכנית. למשל - לא היינו רוצים שבכל פעם שאנו רוצים לכתוב מסמך ב-Word, אנו צריכים לכתוב אותו מהתחלה, כי לא היה ניתן לשמור את העבודה של הפעם הקודמת. נרצה במקרה זה לשמור כל פעם את המסמך שאנו עובדים עליו, ובפעם הבאה לאחזר אותו אל Word, ולהמשיך לעבוד עליו. לשם כך, קיימים רכיבי זיכרון קבועים, כגון Hard-Disk או דיסקטים, שמסוגלים לשמור מידע לאורך זמן, ולהביאו לפי דרישה. המידע נשמר על התקנים אלו בקבצים, שמזוהים על ידי שמם ועל ידי הספרייה בה הם נמצאים.

מסמך זה ידגים עבודה עם קבצים בשפת Visual Basic.

פתיחת קובץ

קריאה וכתיבה לקבצים ב-Visual Basic מתבצעת בצורה פשוטה. השלב הראשון בעבודה עם קבצים הוא "פתיחת הקובץ", ולאחר מכן אנחנו מסוגלים לכתוב או לקרוא ממנו, בהתאם למה שהגדרנו בתהליך הפתיחה.

נניח כי אנחנו רוצים לעבוד על הקובץ c:\test.txt. פתיחת הקובץ נעשית כך:

```
Open "c:\test.txt" For Input As #1
```

הפקודה פותחת את הקובץ לקריאה - כלומר - קריאת נתונים מתוך הקובץ אל התוכנית שלנו.

המילה השמורה For משמשת אותנו לציון דרך העבודה בה אנחנו רוצים לפתוח את הקובץ. במקרה שלנו ביקשנו לפתוח את הקובץ לקריאה בלבד (Input).

סוגי ההרשאות הקיימות:

- **input** – הרשאת העבודה היא לקריאה בלבד מהקובץ
- **output** - כתיבה לתוך הקובץ ללא קריאת נתונים. אם היו בקובץ נתונים קודמים, הם ימחקו.
- **append** - קריאה וכתיבה יחד תוך שימור הנתונים הקודמים שהיו בקובץ. נתונים חדשים יכתבו לסוף הקובץ.
- **binary** - פתיחה של קובץ והתייחסות לנתונים שבו כאל נתונים בינאריים (data).
- **random** – פתיחת הקובץ לגישה רנדומלית – קריאה וכתיבה בכל מקום בקובץ.

הסימון #1 מעיד על שם הקובץ שבו נשתמש בתוכנית. זהו בעצם מזהה שהתוכנית משתמשת בו כדי לזהות את הקובץ הספציפי הנ"ל. אפשר להשתמש לא רק ב-#1 אלא גם בכל שם משתנה אחר עם סימן הסולמית #, לדוגמא #file.

סגירה של קובץ

בסיום העבודה עם קובץ, לצורך שמירה מסודרת של השינויים ושחרור משאבי המערכת, עלינו לסגור אותו. הסגירה מתבצעת על ידי הפקודה Close. לדוגמא:

```
Close #1
```

כאשר #1 זהו שם הקובץ שהגדרנו בעת הפתיחה.

קריאת נתונים מהקובץ

לאחר שפתחנו את הקובץ אנו רוצים לקרוא נתונים. קריאת הנתונים נעשית לרוב על ידי לולאה שקוראת את הקובץ שורה אחר שורה עד שהיא מגיעה אל סוף הקובץ. בכל פעם נכניס את השורה לתוך משתנה, ונבצע עליו את העיבוד הדרוש לתוכנית שלנו.

הקוד הבא פותח את הקובץ, קורא את השורות אחת אחרי השניה ומציג את תוכנו ב-MsgBox:

```
Dim str As String
Open "c:\test.txt" For Input As #1
Do Until EOF(1)
    Line Input #1, str
    MsgBox str
Loop
Close #1
```

תנאי העצירה של הלולאה הוא EOF(1). הפירוש של EOF זה End Of File, כלומר הלולאה ממשיכה עד שמגיע סוף הקובץ #1.

הפקודה

```
Line Input #1, str
```

קוראת בכל פעם שורה מהקובץ, ושמה אותה בתוך המשתנה str.

כתיבה לתוך קובץ

כדי לכתוב בקובץ אנחנו יכולים להשתמש בפונקציות print,write.

השוני ביניהן הוא ש-write מציגה את הכתוב עם "".

דוגמא לשימוש:

```
Open "c:\test2.txt" For Output As #1
Write #1, "Hi"
Close #1
```

ובמקרה של print:

```
Open "c:\test2.txt" For Output As #1
Print #1, "Hi"
Close #1
```

התוצאות שמתקבלות בקובץ:

```
write "Hi" - ע"י
```

```
print Hi - ע"י
```

קריאה וכתיבה בינארית

כתיבה וקריאה בינארית משמשות אותנו לצורך כתיבה של נתונים לקובץ, שמטבעם אמורים להקרא על ידי התוכנית בלבד ולא על ידי בני אדם. קודם, כשיצרנו קובץ טקסט, היה ניתן ללא בעיה לפתוח אותו בפנקס רשימות ולהביט בתוכן שלו. ניתן לעשות זאת גם על קבצים בינאריים, אולם לרוב מה שנשמור בהם יהיה תוכן של משתנים וסוגים אחרים של נתונים שלא יהיו מובנים על ידי קריאה מתוך עורך טקסט.

הפונקציה שמשתמשת אותנו לקריאה בינארית היא הפונקציה `get`. לצורך כתיבה בינארית אנחנו נשתמש בפונקציה `put`.

ע"י `get` אנחנו יכולים לטעון נתונים מקובץ לתוך משתנה. לדוגמא:

```
get #1,,str
```

נשים לב ששני הפסיקים הם בכוונה ולא טעות הקלדה. אם היינו רוצים, בעזרת `get` אפשר לציין גם את נקודת ההתחלה (בבתים), ממנה תתחיל הקריאה.

ע"י שימוש בפונקציה `put` אנחנו יכולים לכתוב נתונים בינאריים לתוך קובץ. לדוגמא:

```
put #1,,str
```

- יש לזכור לפתוח את הקובץ לעבודה במצב בינארי כדי לכתוב או לקרוא מתוכו על ידי פקודות אלו!

כתיבה וקריאה רנדומליים – עבודה עם מבנים

בנוסף לקריאה בינארית עם get אפשר להשתמש בפונקציה כדי לקרוא את הערכים לתוך מבנה כדי לעשות זאת פותחים את הקובץ לגישה רנדומלית ומגדירים את גודל המשתנים שבמבנה. לדוגמא:

```
Private Type SiteInformation
    users As String * 100
    site As String * 100
End Type

Private Sub Form_Load ()
    Dim theSite As SiteInformation
    Open "c:\test.txt" For Random As #1 Len = Len(theSite)
    Get #1, 1, theSite
    MsgBox theSite.site
End Sub
```

ערכי המבנה נקראו מהקובץ.

באופן דומה, ניתן לשמור על ידי put ערכי מבנה לתוך הקובץ.

EOF