

14 שיעור

אוספים, טיפוסים וקבועים

בשיעור זה נסקור אוספים, טיפוסים וקבועים, שהם כלים לאחסון נתונים.

מהו אוסף?

אוסף (Collection) הוא, אובייקט האוסף בתוכו נתונים, הניתן לגשת אליהם באמצעות האינדקס שלהם, בדומה למערכים, או באמצעות המפתח שלהם. אנו הכרנו כבר אוספים רבים בוויזואל בייסיק, מבלי לדעת כי אלו אוספים. למשל כל הטפסים (Forms) בוויזואל בייסיק הם אוספים.

הצהרה על אוסף

מזהירים על אוסף כך:

```
Dim Fruits As New Collection
```

בדוגמא הצהרנו על אובייקט אוסף בשם Fruits. טווח ההגדרה של אוספים, זהה לטווח ההגדרה של משתנים ומערכים (ראו בשיעורים 12,13).

הוספת איברים לאוסף

כדי להוסיף איבר לאוסף, יש להשתמש בשיטה Add, כך:

```
Dim Fruits As New Collection
Fruits.Add "Apple"
```

בשורה הראשונה הצהרנו על אובייקט אוסף חדש בשם Fruits, ובשורה השנייה הוספנו לו איבר חדש, המאכסן את המחרוזת "Apple". האיבר החדש באוסף יקבל באופן אוטומטי את האינדקס הפנוי הראשון. חשוב לזכור שהאינדקס של האיבר הראשון באוסף הוא 1 ולא 0, כמו במערכים (שלא משתמשים ב-Option Base 1). ניתן להוסיף לאיבר המוסף לאובייקט האוסף, גם מפתח (Key), שהוא יוכל להיות מזוהה בעזרתו, במקום בעזרת האינדקס שלו, למשל:

```
Dim Fruits As New Collection
Fruits.Add "Apple", "ID1"
```

בשורה הראשונה הצהרנו על אובייקט אוסף חדש בשם Fruits, ובשורה השנייה הוספנו לו איבר חדש, המאכסן את המחרוזת "Apple" ושהמפתח שלו הוא "ID1". כעת נוכל לקרוא לאיבר המאוכסן במערך באחת משתי הדרכים הבאות: באמצעות האינדקס שלו, למשל כך:

```
Form1.Caption = Fruits(1)
```

שורה זו תשנה את כותרת החלון Form1, לערך המאוכסן באיבר הראשון באוסף. ניתן לקרוא לאיבר באוסף, גם באמצעות המפתח שלו (אם הצהרנו על כזה, בהוספת האיבר), למשל כך:

```
Form1.Caption = Fruits("ID1")
```

שורה זו, תשנה את כותרת החלון Form1, לערך המאוכסן באיבר שהמפתח שלו הוא "ID1".

הורדת איברים מאוסף

כדי להוריד איבר מאוסף יש להשתמש בשיטה Remove, למשל:

1	<code>Dim Processors As New Collection</code>
2	<code>Processors.Add "Intel Pentium 3", "ID1"</code>
3	<code>Processors.Add "AMD Athlon", "ID2"</code>
4	<code>Processors.Add "VIA M3", "ID3"</code>
5	<code>Processors.Remove (2)</code>

בשורה 1 הצהרנו על אובייקט אוסף בשם Processors.

בשורות 2-4, הוספנו שלושה איברים שונים בעלי מפתחות (keys) לאובייקט האוסף.

בשורה 5, הורדנו מאובייקט האוסף את האיבר שהאינדקס שלו הוא 2, כלומר את האיבר השני.

יכולנו להשתמש בשורה 5, במפתח של האיבר במקום באינדקס שלו, על מנת להסיר, למשל כך:

<code>Processors.Remove ("ID2")</code>
--

השיטה Item – החזרת ערך מאיבר באוסף

השיטה Item היא שיטת ברירת המחזל באוסף, למשל במקום לקרוא לאיבר באוסף למשל כך:

<code>MsgBox Processors(1)</code>

יכולנו לקרוא לו כך:

<code>MsgBox Processors.Item(1)</code>
--

הדבר זהה לחלוטין מבחינת התוצאה, וזה רק שאלה של סגנון האם להשתמש ב-Item או להשתמש רק בשם האוסף עם האינדקס או המפתח של האיבר.

מתכנתים רבים כן נוהגים להשתמש ב-Item, למרות שהדבר לא הכרחי, כדי להבדיל אוסף ממערך, כי כשמתכנת רואה את השיטה Item, הוא יודע מיד כי מדובר באוסף ולא במערך, מבלי לראות את ההצהרה על האוסף.

כמובן שניתן להשתמש בשיטה Item גם עם המפתח, כמו שראינו קודם ללא Item, למשל כך:

<code>Dim Processors As New Collection</code>
<code>Processors.Add "Intel Pentium 3", "ID1"</code>
<code>MsgBox Processors.Item("ID1")</code>

התכונה Count

התכונה Count מחזירה לנו את מספר האיברים שיש באובייקט אוסף, למשל:

1	<code>Dim Processors As New Collection</code>
2	<code>Processors.Add "Intel Pentium 3", "Chip1"</code>
3	<code>Processors.Add "Intel Celeron", "Chip2"</code>
4	<code>Processors.Add "AMD Athlon", "Chip3"</code>
5	<code>Processors.Add "AMD Duron", "Chip4"</code>
6	<code>MsgBox Processors.Count</code>

בשורה 1 הצהרנו על אובייקט אוסף בשם Processors.

בשורות 2-5 הוספנו 4 איברים לאוסף, ולכל אחד מהם נתנו גם מפתח.

בשורה 6, תיבת הודעה תציג את מספר האיברים שיש באוסף. במקרה שלנו היא תציג 4.

הבדלים בין מערכים לאוספים

למרות שקיים דמיון רב בין המערכים, אותם הכרנו בשיעור הקודם, לאוספים, עדיין קיימים מספר הבדלים משמעותיים:

1. אוסף גדל וקטן באופן אוטומטי, ולא צריך להצהיר על גודלו מחדש בכל פעם שנוספים או מורדים ממנו איברים, בניגוד למערך (ו מערך דינמי) שאם רוצים להוסיף בו מספר איברים מעבר למה שהוצהר בתחילה, יש להצהיר על כך מחדש בעזרת ReDim. עם אוסף אף פעם לא נקבל הודעת שגיאה שחרגנו מהתחום המוקצה (בניגוד למערך).

2. ניתן לגשת לאיבר באוסף באמצעות האינדקס שלו, בדומה למערך, אך באוסף ניתן לגשת לאיבר גם על ידי המפתח שלו (אם הוצהר כזה). הדבר מאפשר לגשת לאיבר באוסף, מבלי לדעת את מיקומו הנוכחי (מצב שקורה לאחר שהוספנו או הסרנו מספר איברים).
3. כל האיברים באוסף הם מסוג Variant, ולא ניתן להצהיר על סוג איברים אחר שהאוסף יקבל, בניגוד למערך שיכול להיות מערך של כל סוג נתונים.

טיפוסים

מהו טיפוס?

טיפוס (Type) הוא סוג נתונים המוגדר על ידי המשתמש, שיכול לכלול פרט אחד או מספר פרטים. בדרך כלל מאחסנים בטיפוסים נתונים הקשורים אחד לשני. בטיפוס משתמשים באזור ההגדרות בלבד (של הטופס, המודול וכו').

הצהרה על טיפוס

על טיפוס מצהירים באמצעות ההצהרה Type, למשל כך:

```
1 Private Type Computer
2   Display As String
3   Mouse As String
4   Keyboard As String
5   NumberOfProcessor As Integer
6 End Type
```

בשורה 1, פתחנו הצהרה על טיפוס בשם Computer (טיפוס המוצהר בטופס חייב להיות Private, בעוד טיפוס המוצהר במודול יכול להיות הן Private והן Public). בשורות 2-5, כתבנו את הפרטים שיהיו בטיפוס Computer. כל פרט כתבנו מאיזה סוג נתונים הוא. בשורה 6, סיימנו את ההצהרה על הטיפוס.

הדגמת השימוש בטיפוסים

לפריט בטיפוס קוראים על ידי שם הטיפוס, לאחריו נקודה, ואחרי שם הפריט בטיפוס, למשל:

```
Dim Comp As Computer
Comp.Display = "Philips"
```

בשורה הראשונה הצהרנו על משתנה שהוא מסוג Computer, שהוא טיפוס הנתונים עליו הצהרנו קודם. בשורה השנייה, הצבנו לערך Display של המשתנה Comp מטיפוס הנתונים שלנו את המחרוזת "Philips".

בואו נראה עוד דוגמא:

פתחו פרוייקט חדש בויזואל בייסיק, הוסיפו לטופס כפתור פקודה, ואת הקוד הבא:

```
1 Private Type Colors
2   Red As Byte
3   Green As Byte
4   Blue As Byte
5 End Type
6
7 Private Sub Command1_Click()
8   Dim cColor As Colors
9   cColor.Red = 128
10  cColor.Green = 0
11  cColor.Blue = 255
12  Form1.BackColor = RGB(cColor.Red, cColor.Green, cColor.Blue)
13 End Sub
```

- בשורה 1 הצהרנו על טיפוס בשם Colors (צבעים).
 בשורות 2-4 הוספנו לו 3 פריטים שכולם מסוג Byte (ערכים מ-0 עד 255).
 בשורה 5 סיימנו את ההצהרה על הטיפוס על ידי End Type.
 בשורה 7, מתחיל אירוע של לחיצה על כפתור הפקודה Command1.
 בשורה 8, הצהרנו על משתנה בשם cColor שהוא מסוג הנתונים שהצהרנו עליו קודם, Colors.
 בשורות 9-11, הצבנו ערכים לפריטים השונים במשתנה cColor.
 בשורה 12, שינינו את צבע הרקע של הטופס Form1, לערכי הצבע, כפי שקבענו במשתנה cColor.
 בשורה 13, מסתיים האירוע.

קבועים

מהו קבוע?

קבוע (Constant) הוא ערך שלא משתנה במהלך כל התוכנית.

הצהרה על קבוע

ההצהרה על קבוע מתבצעת באמצעות Const, למשל כך:

```
Const PI = 3.14
```

ההצהרה הזו, אומרת לויזואל בייסיק, ליצור קבוע בשם PI ולהציב לתוכו את הערך 3.14.
 אם אנו רוצים שהקבוע יהיה נגיש גם לטפסים ומודלים אחרים יש להצהיר עליו כ-Public למשל כך:

```
Public Const Exponent = 8
```

חשוב להדגיש כי בטופס, קבוע יכול להיות רק Private. כדי להצהיר על קבוע כ-Public יש להצהיר עליו באזור ההגדרות של מודול (Module).

ניתן בעת ההצהרה על קבוע, לקבוע גם את טיפוס הנתונים של הקבוע, למשל:

```
Const Language As String = "Hebrew"
```

כאן הצהרנו על קבוע בשם Language, שהוא מסוג מחרוזת, ואחסנו בתוכו את המחרוזת "Hebrew".
 אך זה לא חובה להצהיר על טיפוס הנתונים, בזמן ההצהרה על קבוע (ואכן רוב המתכנתים לא עושים כך).

את ערכו של הקבוע לא ניתן לשנות במהלך התוכנית. כל ניסיון להציב לקבוע ערך חדש במהלך התוכנית יביא להופעת הודעת השגיאה: *Assignment to constant not permitted*, כלומר שהצבה לקבוע אינה מותרת.

השימוש בקבוע יבוצע כמו השימוש במשתנה רגיל, מלבד העובדה שלא ניתן לשנות את ערכו, למשל:

1	Const PI = 3.14
2	Dim Area As Double, R As Double
3	R = 4
4	Area = PI * R ^ 2

בשורה 1 הצהרנו על קבוע בשם PI שערכו 3.14.
 בשורה 2, הצהרנו על שני משתנים Area ו-R, שניהם מסוג Double.
 בשורה 3, הצבנו למשתנה R את הערך 4
 בשורה 4 הצבנו למשתנה Area (שטח) את התוצאה של הביטוי: $PI * R^2$ (התוצאה צריכה להיות 50.24).

הטיפוס Enum

טיפוס נתונים Enum דומה לטיפוס Type, מלבד שפריטיו של Enum הם קבועים (ולא משתנים כמו ב-Type).

הצהרה על Enum

ההצהרה מתבצעת באמצעות Enum, כך:

1	Private Enum AuthorID
2	Dostoevski = 1
3	Agnon = 2
4	Christie = 3
5	End Enum

בשורה 1 אנו מצהירים על טיפוס Enum בשם AuthorID. בשורות 2-4 אנו מוסיפים שלושה פריטים (קבועים) לטיפוס Enum ומציבים להם ערכים. בשורה 5, מסתיימת ההצהרה על ידי End Enum.

הערות על Enum

אם לא נציב ערך לפריטים ב-Enum הם יקבלו אוטומטית את המספרים 0,1,2... וכך הלאה. אם רושמים לפריט ערך ולפריט אחריו לא רושמים ערך, אז הפריט חסר הערך, יקבל באופן אוטומטי ערך הגדול ב-1 מהערך של הפריט שלפניו, לדוגמא:

1	Private Enum AuthorID
2	Dostoevski = 1
3	Agnon = 8
4	Christie
5	End Enum

הערך של Agnon בשורה 3, נקבע ל-8, ובשורה 4, לא קבענו ערך ל-Christie, ולכן הוא יקבל ערך הגדול ב-1, מזה שלפניו, כלומר הערך של Christie יהיה 9. המגבלות לגבי הצהרת Enum בטופס (כ-Private בלבד) וכו' זהות למגבלות על הצהרת Type.

שימוש בערכים מ-Enum

בערכים מהצהרת Enum משתמשים כך:

```
MsgBox AuthorID.Christie
```

כלומר משתמשים בשם ה-Enum, לאחריו נקודה, ואחר כך שם הפריט (הקבוע) מהצהרת ה-Enum.

שאלה לחזרה

כיצד מסירים איבר מאוסף (Collection)?

1. על ידי השיטה Remove ולאחריה מספר האינדקס של האיבר או המפתח שלו.
2. על ידי שפכטל (מרית בעברית).
3. על ידי מסור.
4. לא ניתן להסיר איבר מאוסף, זהו ההבדל הגדול בין אוספים למערכים ועכברים.