

ארגון נתונים וטבלאות

טבלאות נתונים, רשומות ושדות

ארגון הנתונים ב-Access נעשה בטבלאות המייצגות ישויות שונות (entities) השייכות לאותו עולם מציאותי. מבנה הטבלה נקבע בעת בנייתה או כתוצאה משינוי, ואילו הנתונים הם תכנים שמתוספים, נגרעים או מתעדכנים מבלי לשנות את המבנה במהלך חיי המערכת.

טבלה בנויה משורות, הקרויות רשומות (records), המייצגות פריטים שונים השייכים לאותה ישות. התכונות של הרשומה הן העמודות, הקרויות שדות (fields). לדוגמה אם נתבונן בקטלוג ספרים בספרייה הרי לכל ספר נרצה להגדיר כמה מאפיינים שיופיעו כשדות נפרדים: לדוגמה, שם המחבר, שם הספר, שם ההוצאה, שנת ההוצאה וכו'. כל ספר הוא רשומה והמאפיינים שלו הם השדות תמיד באותו שדה.

כדי שניתן יהיה לבצע עיבוד קל של הנתונים יש לתת את הדעת כיצד לפצל את הנתונים לשדות. אם יהיה ברצוננו למיין או לשלוח נתונים המסודרים לפי שם משפחה רצוי להפריד את שם המשפחה ואת השם הפרטי בשדות נפרדים. איחוד או שרשור של שדות נפרדים בעתיד לצורך שימוש זה או אחר הוא תמיד אפשרי, בשעה שההפרדה היא לעתים בלתי אפשרית לחלוטין.

תכונות של שדה

לכל שדה בבסיס הנתונים יש סדרה של תכונות שנבחרו על ידי המתכנן או שנובעים מברירות מחדל. חלק מהתכונות חובה שתהיינה מוגדרות ואילו חלק אחר יכול להיות לא מוגדר (אופציונלי). כאשר פותחים טבלה לעיצוב מקבלים את חלון דו-שיח כמופיע מטה בתרשים 2. בחלון זה לכל שדה מוקצית שורה המאפשרת את הגדרת התכונות הראשיות.

תכונות שדה ראשיות:

1. לכל שדה חייב להיות שם ייחודי בטבלה
2. לכל שדה חייב להיות אפיון של סוג הנתונים מתוך רשימה אפשרית (ראה בהמשך סוגי נתונים - data types).
3. לכל שדה ניתן להוסיף תאור כחלק מדוקומנטציה לצורך תחזוקה.
4. שדה יכול להיות מפתח בטבלה (יוסבר בהמשך)

תכונות משניות של שדה:

על פי סוג השדה שנבחר נקבע במחצית התחתונה של חלון הדו שיח תכונות נוספות שהן תלויות בסוג הנתון שנבחר לשדה (data type).

סוגי נתונים Data Types

כאשר עומדים על העמודה של סוג שדה בחלון הדיאלוג ניתן לפתוח תיבת ערכים ששדה זה יכול לקבל. תיבה זו רגישה לאות, לדוגמה, על ידי הקשת האות n יבחר סוג שדה נומרי.

1. טקסט (Text) – מאפשר לערכי השדה לקבל כל צרף של תווים כולל ספרות, רווחים, סימני פיסוק. אורך שדה זה בכל מקרה איננו יכול לעלות על 255 תווים.

2. שדה מזכר (Memo) - כמו סוג שדה טקסט אלא שמאפשר כמות של עד 64000 תווים.
3. שדה מספר (Number) – שדה שערכו הוא מספר. סוג המספר שלם, עשרוני ועוד נקבע על ידי תכונת המשנה (ראה טבלת נספח).
4. מספור אוטומטי (AutoNumber) - הוא שדה של שלמים שבו המערכת (ולא המשתמש) קובע את ערכו לכל רשומה. המספור האוטומטי יכול להיות אינקרמנטלי או אקראי.
5. כן/לא (yes/no) – ערכי השדה הם נכון = 1, לא נכון = 0. התצוגה יכולה להיות במלים yes/no או סימון בתיבה.
6. currency- שדה נומרי שמייצג מטבע הן בתצורתו והן במידת הדיוק.
7. שדה תאריך/שעה (Date/Time) - שדה המיועד לתאריכים ועל כן מקבלים רק ערכים שהם קבילים להיות תאריך.
8. אובייקט Ole - שדה שיכול להכיל אובייקטים גדולים כגון תמונות, גרפים ודומיהם.
9. היפר-קישור (Hyper link) – זה הוא שדה המכיל מצביע לכתובת של קובץ, או אתר אינטרנטי.
10. אשף בדיקת מידע (Lookup Wizard) – זה הוא שדה שפותח טבלת ערכים או רשימות כדי להבטיח הכנסת ערכים מדויקים לשדה.

Field Name	Data Type	Description
id	Number	מספר זיהוי של הלקוח
firstname	Text	שם פרטי של המנהל
lastname	Text	שם משפחה של המנהל
zipcode	Number	מיקוד
lastupdate	Date/Time	תאריך עדכון אחרון של פרטים אישיים
picture	OLE Object	תמונה
financial value	Currency	

Field Properties

General

Lookup

Field Size	Long Integer
Format	
Decimal Places	Auto
Input Mask	
Caption	
Default Value	0
Validation Rule	
Validation Text	
Required	No
Indexed	Yes (Duplicates OK)

A field name can be up to 64 characters long, including spaces. Press F1 for help on field names.

תרשים 1 – חלון דו שיח של עיצוב טבלה.

טבלאות מערכת

גם ניהול בסיס הנתונים שהוא אוסף כל ההגדרות המבניות שלנו מתנהל בטבלאות שבאחריות התוכנה. באופן ששמות האובייקטים, המבנה שלהם שמור בטבלאות אלו שעה שהנתונים שמורים בטבלאות הנתונים.

חלוקת הנתונים לטבלאות – קוים מנחים

מאחר שבבסיס נתונים מקובל ישנם מספר סוגי נתונים הקשורים לאותו עיסוק, ואנו מעוניינים לנהל את המערכת בצורה הפשוטה ביותר והחסכונית ביותר. רצוי לפצל את הנתונים על פי סוגיהם. לדוגמה, נניח שאנו חברה הרוכשת מוצרים מספקים ומוכרת ללקוחות ועל כן עלינו לנהל מחירון ומחסן, לקבל הזמנות למוצרים מלקוחות, להוציא תשלומים לספקים וכן חשבוניות ללקוחות, לספק נתונים למערך השיווק על הלקוחות השונים ועוד. על כן נצטרך לתת את הדעת כיצד לאגור את הנתונים השונים.

עקרונות מקובלים למניעת כפילויות:-

א. שדות מחושבים – לא מקובל לאגור נתונים שהם תלויים באחרים. לדוגמה, אם מנהלים מחסן פריטים כאשר לכל פריט יש שדה "מחיר יחידה" ושדה "מספר יחידות" אזי סה"כ ערך הפריט במחסן שהוא תוצר של המכפלה לא יהיה נתון בלתי תלוי ועל כן לא יופיע כשדה בטבלאות.

כאשר נזדקק לערך זה הוא יחושב מקומית ולא יישמר כנתון. על כן הוא תמיד יהיה עדכני ויחסוך בגודל הטבלה וממילא בגודל בסיס הנתונים.

ב. ערכים שחוזרים על עצמם בשדות, פיצול טבלאות וקישור

נתבונן בדוגמה הבאה, חברה מנהלת את כל נתוני ההזמנות בטבלה אחת. הנתונים כוללים בין היתר מספר הזמנה, קוד פריט, שם פריט, שנות אחריות לפריט, כמות מוזמנת, תאריך הזמנה, שם הלקוח, כתובת הלקוח, טלפון של הלקוח, משטר התשלום ללקוח ועוד.

הוכן באקסל:

W	V	U	T	S	R	Q	P	O	N	M	L	K	J	I
ס'קטלוגי	שם הפריט	כמות	מחיר ליח'	סל"כ	מס קטלוגי	שם הפריט	כמות	מחיר ליח'	סל"כ	מס קטלוגי	שם הפריט	כמות	מחיר ליח'	סל"כ
1	משתיק קול	20	1450	29000	15	פילטר אויר	12	340	4080					
1	פילטר אויר	10	340	3400										
3	רובה צייד	5	3480	17300	17	משתיק קול	15	1450	21750					
1	פילטר אויר	6	340	2040	2	אוהל סיידרי	30	250	7500	34	רובה צייד	13	3460	44980

באופן אינטואיטיבי ברור לנו שחלק מהנתונים יחזרו על עצמם מספר רב של פעמים. ללקוח מוגדר היטב תהיה אותה כתובת, אותו טלפון, אותו מספר פאקס וכו', בכל המופעים. לפיכך, הוא מייצג ישות של לקוחות שכדאי להפרידה מההזמנה ורק לקשר אותה באמצעות שדה לטבלת לקוחות. באופן דומה שם הפריט, שנות אחריות לפריט הם זהים עבור אותו קוד פריט. לפיכך, עדיף ליצור טבלה של פריטים שתאחסן את המאפיינים של הפריט. מכאן שנגדיר את הטבלאות הבאות:

- טבלת הזמנה-פריטים
- טבלת הזמנות
- טבלת לקוחות
- טבלת פריטים

כדי שניתן יהיה לשחזר את הנתונים לצרכים שונים, כפי שהבאנו במקור יש צורך ליצור קישור בין ארבע הטבלאות שהוצעו כאן.

בו נתאר לעצמנו שכתובתו של הלקוח השתנתה. אם היינו עובדים בטבלה אחת שטוחה כפי שהוצע בשלב א', היה עלינו לזהות את כל המופעים של הכתובת ללקוח הנדון ולשנות את הערכים של השדות הנ"ל. אם לעומת זאת הספקים מופיעים בטבלה נפרדת הרי לכל לקוח יהיה מופע יחיד ועל כן יהיה צורך בעדכון אחד שייכנס אוטומטית כל אימת שנגזור את נתוני הלקוח. מעבר לחסכון יש גם סיכוי טוב למניעת טעויות.

להלן מבנה הטבלאות כפי שנראה בפתיחתן בתצוגת עיצוב

Table : טבלת לקוחות

Field Name	Data Type	Description
מס לקוח	Number	
שם החברה	Text	
כתובת	Text	
ישוב	Text	
מיקוד	Number	
טלפון	Text	
פקס	Text	
מס עוסק מורשה	Text	

Field Properties

General | Lookup

Field Size: Long Integer

Format:

Decimal Places: Auto

Input Mask:

Caption:

Default Value: 0

Validation Rule:

Validation Text:

Required: No

Indexed: Yes (No Duplicates)

A field name can be up to 64 characters long, including spaces. Press F1 for help on field names.

תרשים 2 תצוגת עיצוב טבלת לקוחות

Table : הזמנה-פריט

Field Name	Data Type	Description
קוד הזמנה	Number	
קוד פריט	Number	
כמויות	Number	

Field Properties

General | Lookup

Field Size: Long Integer

Format:

Decimal Places: Auto

Input Mask:

Caption:

Default Value: 0

Validation Rule:

Validation Text:

Required: No

Indexed: No

A field name can be up to 64 characters long, including spaces. Press F1 for help on field names.

תרשים 3 תצוגת עיצוב הזמנה-פריט

Table : טבלת פריטים

Field Name	Data Type	Description
קוד	Number	
שם פריט	Text	
שם יצרן	Text	
מחיר יחידה	Currency	

Field Properties

General | Lookup

Field Size: Long Integer

Format:

Decimal Places: Auto

Input Mask:

Caption:

Default Value: 0

Validation Rule:

Validation Text:

Required: No

Indexed: Yes (No Duplicates)

A field name can be up to 64 characters long, including spaces. Press F1 for help on field names.

תרשים 4 תצוגת עיצוב – טבלת פריטים

Table : טבלת הזמנות

Field Name	Data Type	Description
מס הזמנה	Number	
קוד לקוח	Number	
תאריך קבלת הזמנה	Date/Time	

Field Properties

General | Lookup

Field Size: Long Integer

Format:

Decimal Places: Auto

Input Mask:

Caption:

Default Value: 0

Validation Rule:

Validation Text:

Required: No

Indexed: Yes (No Duplicates)

A field name can be up to 64 characters long, including spaces. Press F1 for help on field names.

תרשים 5 תצוגת עיצוב – טבלת הזמנות

מפתחות וקשרים בין טבלאות

מפתחות

בכל טבלה רצוי שכל רשומה תופיע רק פעם אחת. במלים אחרות אנחנו רוצים שכל שתי רשומות תבדלנה לפחות בשדה אחד. בדרך כלל ניתן למצוא תת קבוצה של שדות שצירופן הוא יחידני. אותו צירוף מגדיר את הרשומה באופן חד ערכי.

הגדרה: מפתח ראשוני primary key – שדה או צירוף של שדות שמגדירים חד ערכית את הרשומה. לדוגמה, מספר לקוח הוא יחידני ויכול להיות מפתח. בטבלה של הזמנות-פריטים מספר הזמנה איננו מפתח כי לכל פריט בהזמנה אנו מבקשים ליצור רשומה נפרדת. מאידך גם קוד הפריט איננו

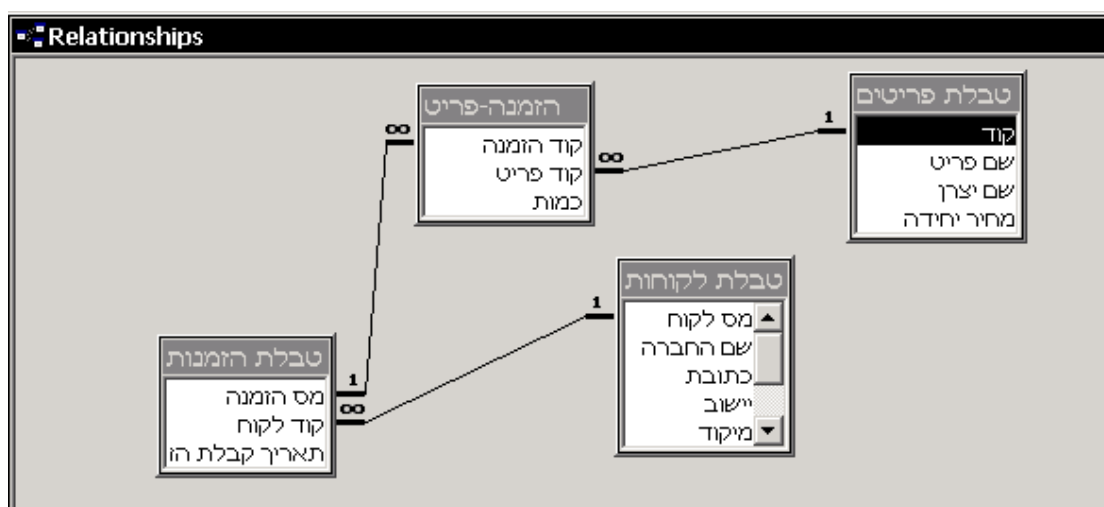
יכול להיות מפתח כי אותו פריט יכול להופיע במספר הזמנות שונות. אולם הצירוף של קוד הזמנה וקוד פריט יכול להיות מפתח.

כעקרון, שדה יכול להיות שדה ריק. יוצאים מכלל זה שדות שהוגדר מפורשות שהם נדרשים ושדות שהם חלק ממפתח.

בתצוגות 3-6 ניתן לראות סימון מפתח בצד השדות המהווים מפתח בטבלה.

קשרים (relationship)

לאחר שבנינו טבלאות והגדרנו שדות מפתח ניתן לקשור בין שדות שבטבלאות שונות. קישור זה יאפשר להציג את כל המידע הנדרש מבלי ליצור טבלאות עם חזרה מרובה על נתונים. לדוגמה, את טבלת הזמנות-פריטים ניתן לקשור לטבלת הזמנות באמצעות קוד ההזמנה ולטבלת הפריטים באמצעות קוד הפריט. את טבלת ההזמנות ניתן לקשור לפרטי הלקוח באמצעות קוד הלקוח.



תרשים 6 תצוגת קשרים בין הטבלאות