

MS SQL Server 2000

סיכום נקודות על ידי דוגמא

נִיר אָדָר

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 לנִיר אָדָר

Nir Adar
Email: underwar@hotmail.com
Home Page: <http://underwar.livedns.co.il>

אנא שלחו תיקונים והערות אל המחבר.

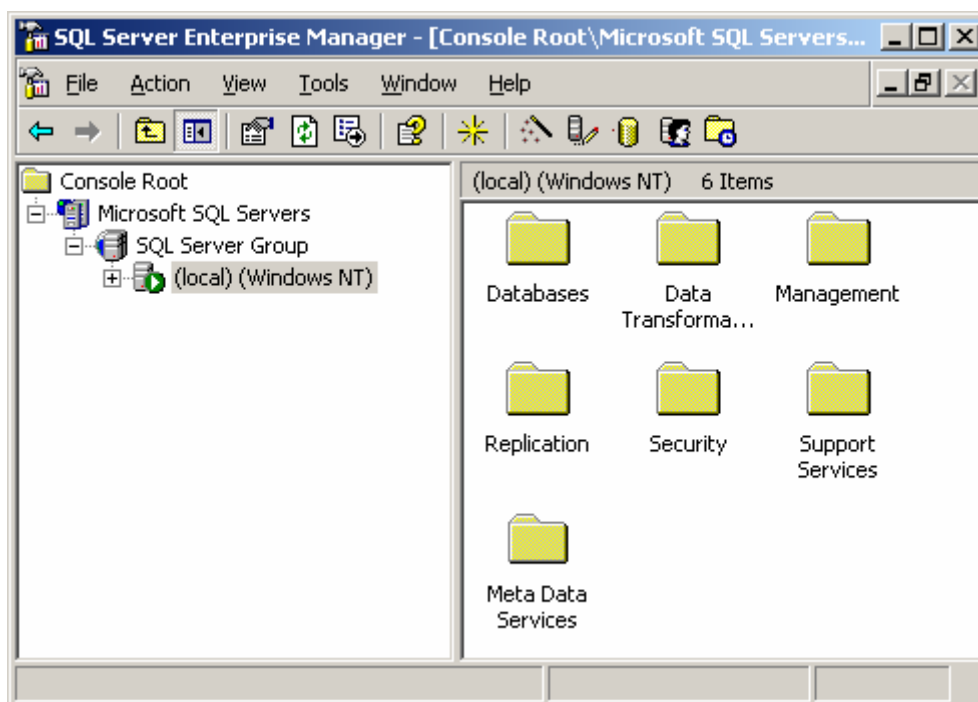
MS SQL Server 2000 - סיכום נקודות על ידי דוגמא

SQL Server 2000 הוא Relational Database Management System – RDBMS. הוא מאפשר יצירה, עדכון וניהול של בסיסי נתונים מקושרים. בסיס נתונים מקושר הוא אוסף של איברים שניתן לגישה ולשינוי ללא צורך בשינוי מבנה הטבלאות בתוך בסיס הנתונים.

SQL Server 2000 תומך בגרסה משלו של שפת SQL המכונה TSQL. השפה מרחיבה את סטנדרט ANSI SQL 92.

התוכנות בהן נתעסק מתוך החבילה שבאה עם SQL Server הן Enterprise Manager ו- Query Analyzer.

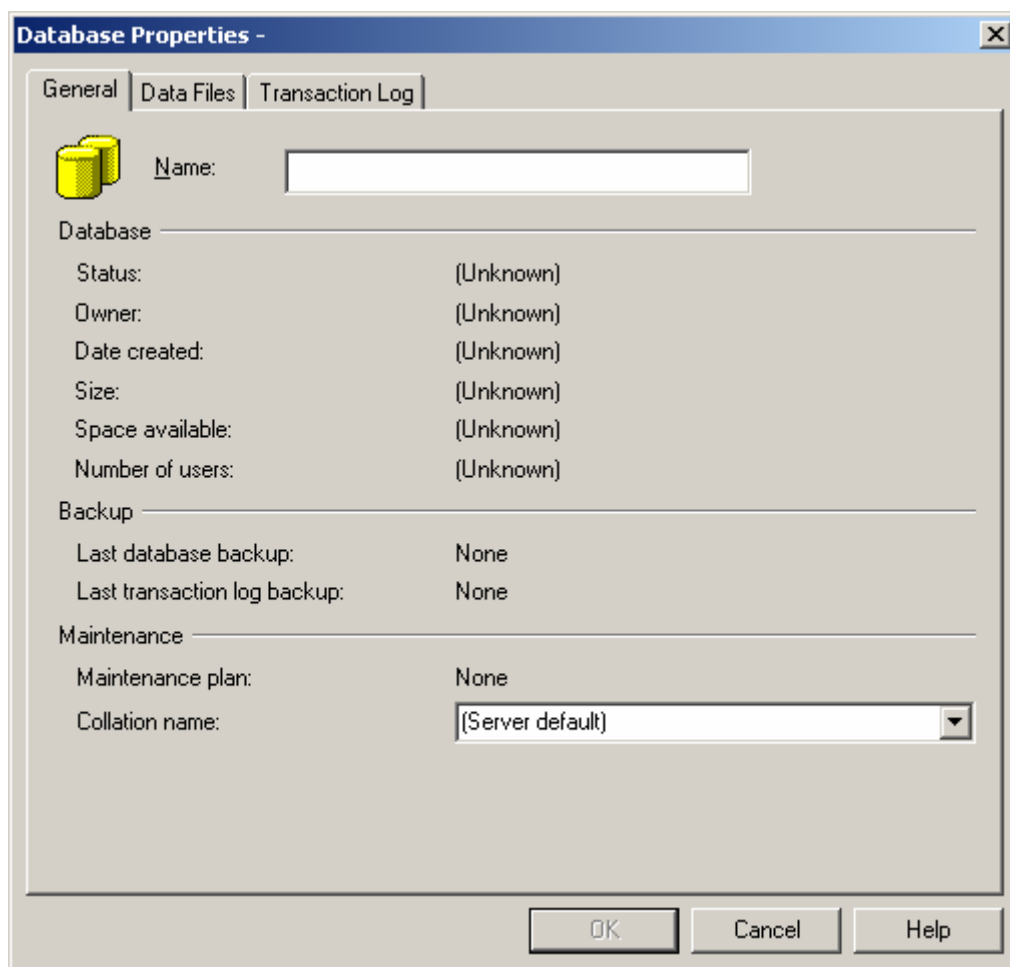
נפתח את Enterprise Manager, ונראה את המסך הבא:



כל התקנה של SQL Server כוללת בסיס נתונים ראשי (master database). טבלה חשובה בו הינה sysdatabases, המכילה את רשימת כל בסיסי הנתונים שנמצאים על השרת.

כאשר אנו רוצים לראות או לשנות את תוכנה של טבלה בבסיס נתונים כלשהו ב-SQL Server נמצא את הטבלה המבוקשת, נלחץ עליה מקש ימני ונבחר Open Table, Return All Rows. נקבל את כל המידע השמור בטבלה, ונוכל לצפות בו ולשנות אותו.

יצירת בסיס נתונים חדש נעשית על ידי לחיצת מקש ימני על Databases ובחירת New Database. נקבל את המסך הבא בו נוכל לבחור את שם בסיס הנתונים הרצוי, ונוכל לקבוע איפה בפועל ישמר בסיס הנתונים.

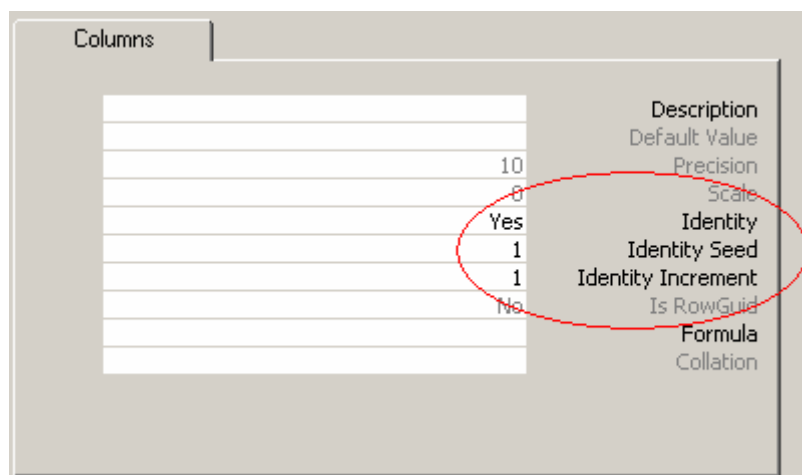


לאחר שיצרנו בסיס נתונים חדש נוכל ליצור טבלה על ידי מעבר אל בסיס הנתונים, בחירת Tables על ידי מקש ימני ובחירת New Table.

יפתח Designer הדומה לזה של Access שיאפשר לנו להגדיר את הנתונים השונים שיישמרו בטבלה.

נניח שנרצה ליצור שדה שיהווה מזהה לכל איבר, כך שלכל איבר יוקצה מזהה יחודי. נעשה זאת בצורה הבאה: ניצור שדה, שנכנה כאן ID. נבחר את סוג השדה להיות int – מספר שלם. כמו כן נסמן שערכו של משתנה זה אינו יכול להיות NULL.

בתכונות נבחר את Identity להיות Yes, את ה-Seed להיות 1 ואת ה-Increment להיות 1.



בצורה כזו לכל איבר שנכנס יהיה מזהה יחודי שיגדל בכל פעם ב-1, כאשר הערך ההתחלתי של האיבר הראשון גם הוא 1.

בשביל הדוגמא נוסיף שני שדות נוספים לטבלה: Name מסוג varchar בגודל 20, ושדה age מסוג int.

נראה כיצד יוצרים את בסיס הנתונים ואת הטבלה שוב, הפעם דרך **SQL Query Analyzer** – בעזרת פקודות במעין שפת תכנות.

נסגור את Enterprise Manager ונפתח את Query Analyzer. מוצג לפנינו בתחילה מסך Login. במקרה והשרת שלנו הוא שרת חדש, הרי ששם המשתמש ההתחלתי הוא sa והססמא היא ססמא ריקה.

יפתח חלון בו נוכל לכתוב פקודות. נשים לב ששפת ה-TSQL איננה Case-Sensitive, כלומר הביטוי DROP יהיה זהה לביטוי Drop למשל.

מימוש בשפת TSQL של יצירת בסיס הנתונים.
נתחיל במחיקת בסיס הנתונים, במקרה והיה קודם:

```
USE MASTER

GO

-- If the database already exists, drop it
IF EXISTS (SELECT * FROM sysdatabases WHERE name='ExampleDB')
    DROP DATABASE ExampleDB

GO
```

הפקודה GO אומרת לשרת להריץ את הפקודות החל מה-GO הקודם ועד אותה נקודה.
כל קוד המופיע לאחר – הינו הערה והשרת מתעלם ממנו.
אנו בודקים את הטבלה שאנו רוצים ליצור כבר קיימת ואם כן מוחקים אותה.

כעת ניצור את בסיס הנתונים החדש:

```
-- Create the MyDatabase1 database
CREATE DATABASE ExampleDB
GO

USE ExampleDB
GO
```

ויצירת הטבלה תבוצע כך:

```
-- Create the table in ExampleDB
CREATE TABLE tbl_myTable
(
    Id INT NOT NULL PRIMARY KEY IDENTITY(1,1),
    Name VARCHAR(20) NOT NULL,
```

```
    age INT NOT NULL DEFAULT 18
)
GO
```

כדי להריץ את הפקודות נלחץ על F5 או על הכפתור Play.
הקוד ירוץ, כאשר בחלון התחתון נראה את תוצאות הפעולות שביצענו.

הוספת ערכים לדוגמא לטבלה:

```
USE ExampleDB
GO
INSERT INTO tbl_myTable (Name, age) VALUES('Red', 9)
INSERT INTO tbl_myTable (Name, age) VALUES('Blue', 14)
INSERT INTO tbl_myTable (Name, age) VALUES('Green', 24)
INSERT INTO tbl_myTable (Name) VALUES('Orange')
INSERT INTO tbl_myTable (Name, age) VALUES('Purple', 19)
INSERT INTO tbl_myTable (Name) VALUES('Yellow')
GO
```

הפעלת שאילתת SQL על הטבלה וקבלת התוצאות שלה:

```
USE ExampleDB
GO

SELECT * FROM tbl_myTable ORDER BY Name ASC
GO
```

EOF