שיעור 07-04-2019

* ADO.NET
* Active Data Object

טכנולוגיה המאפשרת לנו להתחבר לבסיס נתונים לטובת ביצוע שאילתות ומניפולציות בצד שרת.

שיטות התחברות ב-ADO.NET:

1. **שיטת ה-connected layer** – בשיטה זו בסיס הנתונים מחובר כל הזמן לאפליקציה. בשיטה זו בד"כ כותבים פרויקט dll שבו כותבים את כל הלוגיקה של ה-DAL.
2. **שיטת Disconnected Layer** – שיטה ויזואלית לחלוטין המאפשרת לבצע שימוש באובייקטים מובנים. אוביקט האב מוכר כאובייקט בשם DataSet כאשר בכל DataSet קיימים מספר DataTable
3. **Entity Framework** – שיטה זו מוכרת כשיטת ORM – Object Relational Mapping – את השיטה הזו ניתן לחלק למספר תתי שיטות פיתוח:
   1. Db First
   2. Model First
   3. Code First

**שיטת connected layer:**

מחלקות שימושיות:

using System.Data

using System.Data.SqlClient

1. מחלקת SqlConnection – מחלקה המאפשרת לנו להתחבר לבסיס הנתונים תוך שימוש בקובץ ה-app.config או ה-web.config וזאת תוך שימוש במחלקת ConfigurationManager
2. SqlException – מחלקה המאפשרת לנו לקבל מידע על סוגי טעויות שונות ב-SQL
3. SqlCommand – מחלקה המאפשרת לנו לכתוב את השאילתה ולהריצה.
4. SqlParameter – מחלקה המאפשרת לנו לשלוף ולהעביר קלט/פלט של פרמטרים לבסיס הנתונים.
5. SqlDbType – מחלקה המאפשרת לנו להגדיר את ה-type הייעודי של המשתנה.

**שיטת התחברות Entity Framework**

EF הינה טכנולוגיה המבוססת על ORM (Object Relational Mapping)

טכנולוגיה זה מאפשרת לנו לבצע את הפעולות על בסיס הנתונים.

EF הוצג ב NET 4(2010) ובגרסה 4 בזמנו.

כיום הגרסה העדכנית הינה גרסה 6.2.2 ובעדכון של Visual studio 2019 הגרסה תהיה 7.

EF הינה שכבת טכנולוגיה מעל ADO NET API

C# code

IEnumerable

Linq Query

Entity Sql

Object Services

Entity Client Provider

EDMX

ADO.NET API Provider

DB

כאשר יוצרים EF קיימים 3 שיטות פיתוח:

1. DB First – כאשר קיים בסיס נתונים פיזי, ניתן לייצר קובץ edmx (entity data model xml) שמאפשר לבצע את ההמרה של כל טבלת SQL לאובייקט ב-C# - בשיטה זו מתקיימים 3 תהליכים להלן:
2. נוצר קובץ עם סיומת csdl – קובץ המודל הקונספטואלי – קובץ זה אחראי ומגדיר את הקשרים בין ה-entities
3. נוצר קובץ עם סיומת msl – קובץ המיפוי אחראי להציג את המיפוי של הטבלאות קשרי הגומלין והמפתחות
4. נוצר קובץ ssdl – קובץ זה מציג לנו את המודל הפיזי של בסיס הנתונים.

יתרון של שיטה 1 – פיתוח מהיר וברור

חסרון מהותי – באפליקציות עסקיות מרובות משתמשים ובקשות לבסיס הנתונים תהיה קיימת איטיות מאחר וכל בקשה עוברת את שלושת השכבות.

(מייקרוסופט הודיעו כי בגרסת VS 2019 לא יהיה מודל ויזואלי – בגרסת 2019 זה יהיה EF 7)

1. Model First – בשיטה זו (שאיננה שימושית בתעשייה) יוצרים מודל EF ריק לחלוטין ומתחילים לייצר את הטבלאות ואת קשרי הגומלין בינהם. בסופו של המודל ע"י לחיצה עם קליק ימני על המודל ניתן לייצר בסיס נתונים חדש מהמודל שיצרנו.

יתרון של שיטה 2 – אפשרות מידול ארכיטקטורת בסיסי נתונים בכלי ויזואלי וללא יידע בתכנות.

חסרון – אף אחד בתעשייה לא משתמש בזה

1. Code First – בשיטה זו הידועה כשיטת POCO (Plain old CLR Objects) יוצרים מחלקות ומאפיינים ובעזרת המחלקות והמאפיינים, יחד עם מחלקות של Entity יוצרים את מבנה (סכימה) בסיס הנתונים. שיטה זו היא השיטה הנפוצה ביותר ממספר טעמים:
   1. לא קיימת ויזואליזציה (זאת אומרת לא נוצר ולא יוצרים קובץ edmx)
   2. שיטה שמאוד נוחה בכתיבת תשתיות ובפרויקטים חדשים.
   3. קיימת אפשרות לעבוד מול Linq to Entities
   4. עבודה ישירה מול ADO NET API

חלק מחסרונות של שיטה מספר 3, ברגע שיש שינויים במבנה הנתונים יש צורך לבצע פעולות מיגרציה למינהן.