

# Advanced C#

# נושאים

- Extension methods
- Reflection
- Linq
- Linq to xml
- Serialization
- File system watcher

# Extension method

- הוספת שיטות וproperties למחלקות או ממשקים.
- ניתן להשתמש ישירות על אובייקט של המחלקה.
- הגדרה של שיטה סטטית בתוך מחלקה סטטית.
- שינוי קוד קיים ללא סיכון פגיעה במימוש המקורי.

# Reflection

- עבודה עם מחלקות באופן דינמי בזמן ריצה.
- נוכל להשתמש בכל מחלקה מתוך חיפוש על בסיס שם או על ידי הטייפ אותו אנחנו בודקים.
- נוכל לאפשר קבלת נתונים בצורה דינמית ולהתייחס לכל אובייקט בצורה שונה.
- ניתן לקרוא פרוייקט שלם (קבצי DLL) ולקבל את כל המחלקות בהם נוכל להשתמש.
- ניתן להשתמש בproperty, שיטות, field.

# Linq

- סינטקס פשוט לביצוע שאילתות מורכבות על בסיס נתונים.
- חלק מהשאילתות הקיימות בלינק הינם "עצלניות".
- ניתן לשרשר שאילתות אחת לאחר השנייה.
- סינטקס של פונקציות ושאילתות (בדומה לSQL).
- ניתן לבצע שאילתות על כל אובייקט שיורש מהמחלקה IEnumerable.
- ניתן לבצע שאילתות על בסיסי נתונים כמו Entity Framework, SQL, MongoDB ועוד.
- שימוש בפונקציות לביצוע שאילתות על בסיסי הנתונים (לרוב שיטות אנונימיות).

# שאלות - Linq

```
var foundStudents = students  
    .Where(student => student.Courses!.Contains("Algorithms"));
```

• שאלות בסיסיות:

• Where

• SelectMany, Select

• OrderBy, OrderByDescending

• GroupBy

• .Count, Average, Sum

# XML(Extensible Markup Language)

- תקן קידוד וייצוג נתונים במחשב.
- רכיב מידע בנוי מתגית (tag), תכונות (attributes) ותתי רכיבים.

- דוגמה לקובץ XML:

```
<?xml version="1.0" encoding="utf-8"?>
<Students>
  <Neo ID="b8455e79-c969-4080-a7d1-108c5b3d9fee">
    <Name>Neo</Name>
    <Avg>85</Avg>
    <PhoneNumber>0512345678</PhoneNumber>
  </Neo>
  <Morpheus ID="6ba3477b-d3bf-459c-adba-95e8c5bfcf3c">
    <Name>Morpheus</Name>
    <Avg>70</Avg>
    <PhoneNumber>0512345678</PhoneNumber>
  </Morpheus>
  <Agent_Smith ID="6089eadd-4089-4505-8b11-056ab4b7cc31">
    <Name>Agent Smith</Name>
    <Avg>85</Avg>
    <PhoneNumber>0512345678</PhoneNumber>
  </Agent_Smith>
</Students>
```

# Linq xml

- ניתן להמיר אובייקט לקובץ XML בעזרת LINQ.
- ניתן להפעיל שאילתות לינק על קבצי XML לצורך יצירה, עדכון וביצוע שאילתות לחיפוש.



# Serialization

- סריאליזציה היא מעבר בין אובייקט מסוג אחד לאחר, לרוב לצורך אחסון במסדי נתונים והעברה של נתונים.
- לרוב נבצע סריאליזציה לאובייקטים כמו JSON, BSON, XML, String.
- בעת יצירת מחלקות נוכל לתת attribute לproperties של המחלקה לצורך ביצוע סריאליזציה נכונה.

# File system watcher

- מחלקה למעקב אחר תיקיות.
- מאפשר למפתחים לעקוב אחר שינויים במערכת הקבצים בזמן אמת.
- מחלקה חשובה לאפליקציות אשר צריכות מעקב על מערכת הקבצים לצורך סנכרון קבצים, מעקב אחר יומנים וכו'.
- יצירת אובייקט למעקב אחר שינויים (יצירה, מחיקה, שינוי שם, שינוי תוכן ועוד).