בס"ד תשפ"ג סמסטר א'

**מיני פרויקט במערכות חלונות**

*שלב 5*

מטרות השלב

בשלב זה יתבצע עדכון שכבות הנתונים והלוגיקה למימוש מתודות ה CRUD בעזרת LINQ בלבד במקום לולאות בכל מקום שניתן. פירוט בהמשך.

בשלב זה נשלים את בניית הממשק הגרפי עבור המערכת שלנו.

עיצוב הממשק הגרפי יכול להתבסס על טכנולוגיות שונות (חלונות, עמודים (Page), חלון יחיד עם לשוניות או "מדפי הרחבה", וכל עיצוב אחר שתבחרו).   
ההנחיות יתייחסו לטכנולוגית חלונות אבל אתם מוזמנים להתאים את עבודתכם לעיצוב שבחרתם.

### **דרישות כלליות של מעבר לעיבוד נתונים בעזרת LINQ:**

**לאחר שלמדנו תחביר LINQ לעבודה מול אוספים, נרצה להעביר כל מקום בקוד שעובד מול אוסף ומשתמש בלולאת foreach, למתודות ו-\או תחביר LINQ (כמובן היכן שאפשר).**

בתחביר LINQ ישנן שתי אפשרויות שלמדנו למימוש:

א. בעזרת מתודות הרחבה

ב. בעזרת תחביר שאילתות

**בפרויקט הדרישות הן כדלקמן:**

* בגרסה הסופית אין להשתמש בלולאת foreach במקום שניתן להשתמש ב-LINQ
* ניתן להשתמש בפורמט שאילתות ו-\או בפורמט זימון מתודות הרחבה (רצוי בשרשור)
* עם זאת, חובה להשתמש בשני הפורמטים, לפחות 4 מופעים מכל סוג!
* חובה להשתמש בביטויי למבדה במקומות המתאימים.
* ב-LINQtoObject חובה להשתמש לפחות פעם אחת ב:
  + let
  + select new
  + קיבוץ (grouping)
  + מיון

### 

### **דרישות כלליות של שכבת התצוגה PL:**

1. בשכבת התצוגה חובה לתפוס את כל החריגות ולהציג למשתמש הודעה על הבעיה
2. ההודעה על הבעיה חייבת להיות ברורה ומובנת למשתמש של המערכת (ולא למתכנת) - זאת אומרת המטרה היא **לא** להציג את הודעת החריגה ושמות המשתנים וכדומה אלא להציג הודעה אינפורמטיבית ברמה של משתמש המערכת
3. בחלונות של הוספה - לאחר ביצוע מוצלח של ההוספה - חלון הוספה צריך להיסגר אוטומטית וחלון הרשימה חייב להתעדכן אוטומטית
4. בחלונות עדכון - לאחר ביצוע מוצלח של העדכון - חלון הרשימה חייב להתעדכן אוטומטית

**חובה** להגדיר תכונות מתאימות בקוד האחורי של **כל אחד מהחלונות** על מנת להחזיק את הנתונים המוצגים ו-או המתעדכנים. יש להגדיר את התכונות כך שהשמת אובייקטים בהם תתעדכן התצוגה אוטומטית בצורה דינמית. זאת אומרת - התכונות יוגדרו באחת משתי קונפיגורציות:

* **אפשרות 1**: במחלקה יוגדר אירוע של ממשק INotifyPropertyChanged, וב-set של התכונות האלה יופעל האירוע בהתאם
* **אפשרות 2**: התכונות יוגדרו כתכונות תלות (Dependency Properties)

**חובה** להשתמש ב-**Data Binding** **באופן מלא** (זאת אומרת - קישור תכונות הפקדים לנתונים יתבצע אך ורק מ-**xaml**) **בכל אחד מהחלונות** על מנת לסנכרן את התצוגה עם הנתונים. **אסור** לקשר את הנתונים בקוד האחורי (code behind) - יש לצמצם את הקוד האחורי למינימום הכרחי לטיפול בקלט מהמשתמש ובאירועים אחרים של WPF.

* בשכבה זו **לא תתבצע לוגיקה** של הפרויקט למעט בדיקת תקינות קלט בסיסית (פורמט, תווים חוקיים וכו')
* בכל פעולה המבוצעת מ-GUI יש לשלוח לכל היותר בקשה אחת ל-BL
* יש להשתמש במגוון רכיבים של WPF כפי המופיע סילבוס:
  + פקדים בסיסיים
  + פקדים מרובי נתונים ( List, ComboBox ועוד)
  + תסדירים (Layout) למיניהם (Grid, StackPanel, ועוד)
  + שימוש במשאבים (Resources)
  + קישור (Binding) בין תכונות תצוגה שונות וקישור של תכונות תצוגה לנתונים בקוד האחורי - בכמה דרכים עם עדכון חד כיווני ודו-כיווני
  + שימוש ב-**DataContext** בצורה היררכית - עבור קישור לנתונים (יש לקשור בתוך xaml את תכונת DataContext של החלון לאובייקט החלון עצמו)
  + כדאי להשתמש ב-ObservableCollection עבור החזקת הרשימות ב-PL
  + יש להשתמש בהמרות בקישור נתונים IConverter על מנת לחסום קלט בפקדים בצורה דינמית או להסתיר פקדים בצורה דינמית
* **אפשריות בונוס**:
  + שימוש בפונקציונליות נוספת של WPF שאיננה חובה בסילבוס של הקורס, למשל:
    - טריגרים (trigger - שלושה סוגים - טריגר תכונה, טריגר נתונים, וטריגר אירוע)
    - תבניות עיצוב (DataTemplate) בכללי ולתבניות אלמנט בפקד מרובה נתונים בפרט
    - סגנון (Style), ערכות נושא (Theme)
    - תבניות פקד (ControlTemplate)
    - טרנספורמציות של פקדים (Transform)
    - תכונות מצורפות (attached properties)
    - גרפיקה (drawing, shapes)
    - פקודות והתנהגויות (commands and behaviors)
    - ועוד ועוד ועוד - לפי הזמן והרצון של הסטודנטים
  + שימוש בישויות תצוגה PO וב-data binding מלא דרך ישויות אלה (כנ"ל):
    - יצירת ישויות PO מתאימות - בלי להוסיף כל קוד לישויות של BO
    - בישויות PO - עבור אוספי אובייקטים ניתן (וכדאי) להשתמש ב-ObservableCollection
  + לא יהיה שיתוף מידע בין החלונות - העברת המידע תתבצע בעזרת העברת ארגומנטים עבור הפרמטרים של בנאי החלון, או העברת מתודה לבנאי בפרמטר מסוג דלגט
  + בונוס שיפור ארכיטקטורה של שכבת תצוגה (PL MVVM)