## Anonymous Methods

מתודות אנונימיות מאפשרות כתיבת מתודות בלי שם.

עד עכשיו כשעבדנו עם delegates הגדרנו מתודה, ואז הצבענו עליה כשיצרנו מופע. (בתהליך האינסטנטינציה - Instantiation). כלומר, העברנו כפרמטר את שם הפונקציה בה השתמשנו כשהגדרנו את המופע של ה delegate (בשלב 2 intantiation).

לדוגמה:

// Declare a delegate.

delegate void Del(int x);

// Define a named method.

void DoWork(int k) { /\* ... \*/ }

// Instantiate the delegate using the method as a parameter.

Del d = obj.DoWork;

החל מגרסה 2 של C# מתאפשר ליצור מופעים של delegates, ומיד להגדיר את קטע הקוד שיורץ כשיקראו לפונקציה (כשיעשו invocation). זה פשוט מקצר לנו את הקוד וחוסך לנו הגדרה בנפרד של המתודה.

// Declare a delegate.

delegate void Del(string name);

// Instantiate Del by using an anonymous method.

Del del3 = delegate (string name)

{ Console.WriteLine($"Notification received for: {name}"); };

שימו לב, שבדוגמה הזו בהגדרת המופע מיד כתבתי בלוק קוד להפעלה עליה אני מצביע ולא העברתי שם של מתודה שהגדרתי במקום אחר. אני משתמש בזה כאשר מיותר לי להגדיר מתודה בנפרד, ואני צריך רק את המצביע למתודה.

להגדרת מתודה אנונימית, משתמשים במילת המפתח delegate (אותה מילת מפתח בה אנו משתמשים להגדרת ה delegate), ואז בסוגריים את הפרמטרים שלנו ואת בלוק הקוד.

ועוד דוגמה:

MyFirstDelegate first1 = delegate () { Console.WriteLine("Shalom"); };

MyFirstDelegate first2 = delegate () { Console.WriteLine("Another function called with same delegate"); };

בשביל להבין מתי נשתמש בזה "בחיים האמיתיים" ניזכר במתודות כמו Sort או Where שהוגדרו במחלקה List. אנחנו צריכים להעביר לה מתודה שעל פי המתודה הזו מתבצע ה Sort. במקום להגדיר את המתודה בנפרד ולהעביר אותה, הרבה יותר נוח במהלך הכתיבה שלנו, פשוט לכתוב את המתודה שמגדירה את הקריטריון של ה Sort או הפילטר.

## תרגילי כיתה ובית

לטובת תרגול ניתן לבצע את התרגילים שהיו לנו ל Delegates הפעם עם Annonymus Methods ועם Lambda Expressions.