

## מתי משתמשים ב-base

נתונות המחלקות הבאות:

<pre>public class Person {     protected string name;     protected int yearBorn;     private int id;      public Person (string name, int year, int id)     {         this.name = name;         this.yearBorn = year;         this.id = id;     }      // פעולות get ו-set לכל תכונה      public virtual int CalcAge (int yy)     {         return yy - this.yearBorn;     }      public override string ToString ()     {         string str = "name: " + this.name +             "age: " + this.CalcAge();         return str;     } }</pre>	<pre>public class Student : Person {     private int cita;      public Student (String name, int year, int id, int cita)         : base (name, year, id)     {         this.cita = cita;     }      // פעולות Get ו-Set לכל תכונה      public override int CalcAge (int yy)     {         int age = base.CalcAge (yy);         if (age &lt; 6) return 1;         if (age &lt; 12) return 2;         if (age &lt; 18) return 3;         return 4;     }      public override string ToString ()     {         return base.ToString() + "cita: " + this.cita;     } }</pre>
---	---

ניתן להשתמש ב-base בשני המקרים הבאים:

### 1. בפעולה הבונה של תת-המחלקה.

הפעולה הבונה מקבלת את כל התכונות (כולל אלו שמוגדרות במחלקת העל).

בפעולה הבונה, בשורת החתימה מפעילים את הבנאי של מחלקה העל: (תכונות מחלקת העל) base ובתוך הפעולה מאתחלים את שאר התכונות של המחלקה הנוכחית (תת המחלקה).

דוגמה: הבנאי של המחלקה Student שהיא תת-מחלקה של Person:

```
public Student (string name, int year, int id, int cita) : base (name, year, id)
(
    // הפעלת הבנאי לאתחול התכונות של מחלקת העל Person
    this.cita = cita; // אתחול התכונות המוגדרות במחלקה הנוכחית
)
```

2. פנייה לפעולה המוגדרת במחלקת העל/הבסיס (virtual) אבל קיבלה הגדרה מחדש (override) במחלקה הנוכחית: (שם-פעולה. base) (ההגדרה מחדש "מסתירה" בפני המחלקה את ההגדרה המקורית). (אם תת-מחלקה/מחלקה נגזרת הגדירה מחדש פעולה שכבר מוגדרת במחלקת העל/מחלקת הבסיס, והיא מעוניינת להפעיל את הפעולה של מחלקת הבסיס.) לדוגמה:

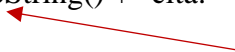
- Student מגדירה מחדש את CalcAge(). היא רוצה להשתמש בפעולה CalcAge() שקיבלה מ-Person ולהוסיף משלה. לכן היא תפנה ל- base.CalacAge(...)

```
public override int CalcAge (int yy)
{
    int age = base.calcAge (yy);
    if (age < 6)    return 1;
    if (age < 12)  return 2;
    if (age < 18)  return 3;
    return 4;
}
```



- Student מגדירה מחדש את הפעולה ToString() ולכן הוא תשרשר למחרוזת את base.ToString()

```
public voerride string toString ()
{
    return base.ToString() + "cita: " + this.cita;
}
```



כל המחלקות יורשות מהמחלקה Object (ממחלקת הבסיס הקיימת בשפה).  
 ב- Object הוגדרו פעולות וירטואליות (ניתנות למימוש מחדש) כמו: ToString() ו- Equals()  
 ולכן כל מחלקה שתממש את שתי הפעולות תגדיר בחתימת הפעולה override (הגדרה מחדש).

#### דגשים:

- base אינו מוגדר מחוץ למחלקה המגדירה את העצם (אין להשתמש ב- base בתכנית או בפעולה חיצונית).
- לא ניתן לבצע של base של base - ( ToString() base. base ) ייתן הודעת שגיאה, גם אם קיימת פעולה כזו בהיררכיה.