

Polymorphism By Abstract Class & Abstract Method

בדוגמה להלן 3 מחלקות : מחלקת בסיס סטודנט ו שתי מחלקות המקיימות יחסי ירושה. בכל מחלקה יש מימוש שונה של אותה מתודה בשם MyNameIs.

```
abstract class Student
{
public abstract void MyNameIs ();
}
class Shani :Student
{
public override void MyNameIs ()
{
System.Console.WriteLine("Student Name: My Name Is Shani.");
}
}
class Doron : Shani
{
public override void MyNameIs ()
{
System.Console.WriteLine("Student Name: My Name Is Doron.");
}
}
class MainClass
{
static void Main(string[] args)
{
Student[] std = new Student[2];

std[0] = new Shani ();
std[1] = new Doron ();

for (int i=0; i<2; i++)
std[i].MyNameIs ();
}
}
```

**abstract class** : לא ניתן ליצור ממחלקה זו עצמים המחלקה חייבת להידרס  
**abstract function** : פונקציה ללא מימוש החייבת בדריסה  
אחרת נקבל שגיאה:  
does not implement inherited abstract member

יצירת אובייקט מערך מסוג Student (מחלקת בסיס) 2 תאים

יצירת אובייקטים מסוג Shani ו Doron

ריצה פשוטה על המערך וקריאה למתודה MyNameIs כתוצאה מדריסת הפונקציות כל אובייקט יממש אחרת את אותה מתודה = רב - צורניות = polymorphism

```
Student Name: My Name Is Shani.
Student Name: My Name Is Doron.
Press any key to continue
```

Virtual Function : פונקציה שניתנת לדריסה ויכולה להכיל מימוש  
Abstract Function : פונקציה שחובה לדרוס ואינה מכילה מימוש