

---

# Introduzione a C#

---

Prof. Francesco Accarino  
IIS Altiero Spinelli Sesto San Giovanni

# C#: Strumento Principe per .net

- Primo linguaggio orientato alle componenti della famiglia C/C++
- Tutto è DAVVERO un oggetto
- Unisce potenza, semplicità e chiarezza di C++, Java, VB...
- Completamente nuovo, non ha vincoli con il passato

# C#: Strumento Principe per .net

- **Indipendente dalla piattaforma e scalabile** (capacità di un sistema di "crescere" o "decreocere" (**aumentare o diminuire di scala**) in funzione delle necessità e delle disponibilità)
- “Nato con” e “cucito addosso” al .NET Framework
- Può essere integrato nelle pagine web
- **Completamente Type Safe** (sicurezza rispetto ai tipi è la misura con cui un **linguaggio di programmazione** previene o avvisa rispetto agli *errori di tipo.*)

---

# Tutto è un oggetto

- In C# tutti i tipi reference, sono derivati dal tipo base `System.Object`.
- Ogni oggetto eredita i metodi base da `System.Object`

... ma proprio **OGNI** tipo...

# Tutto è un oggetto

```
using System;
class Test
{
    static void Main()
    {
        Console.WriteLine(3.ToString());
    }
}
```

ToString restituisce una stringa che rappresenta il valore dell'oggetto

# Tutto è un oggetto

```
using System;
public class MyClass
{
    public static void Main()
    {
        object a;
        int i = 123;
        a = (object)i; // un esempio di boxing
        Console.WriteLine(a);
        Console.WriteLine(a.GetType());
        Console.WriteLine(a.ToString());
        Console.WriteLine();
    }
}
```

Output

123

System.Int32

123

Il boxing è il processo di conversione di un tipo **valore** nel tipo **object** o in qualsiasi tipo di interfaccia implementato dal tipo valore. L'unboxing consente invece di estrarre il tipo di valore dall'oggetto. Nell'esempio viene eseguito il boxing della variabile integer i assegnandola all'oggetto a.

# Boxing ed Unboxing dei Value

```
using System;
class Test
{
    static void Main()
    {
        int i = 1;
        object o = i; // boxing
        // verifica che o sia davvero un int
        if (o is int)
        {
            Console.WriteLine("O is an int");
        }

        int j = (int)o; // unboxing
    }
}
```

---

# Hello, World: C#

```
//Permette riferimenti semplificati al namespace System
using System;

//Questa classe esiste solo per ospitare la funzione entry point dell'applicazione
class MainApp
{
    // Il metodo statico Main è l'entry point dell'applicazione
    public static void Main()
    {
        // Scrive il testo sulla console
        Console.WriteLine("Hello World using C#!");
    }
}
```



---

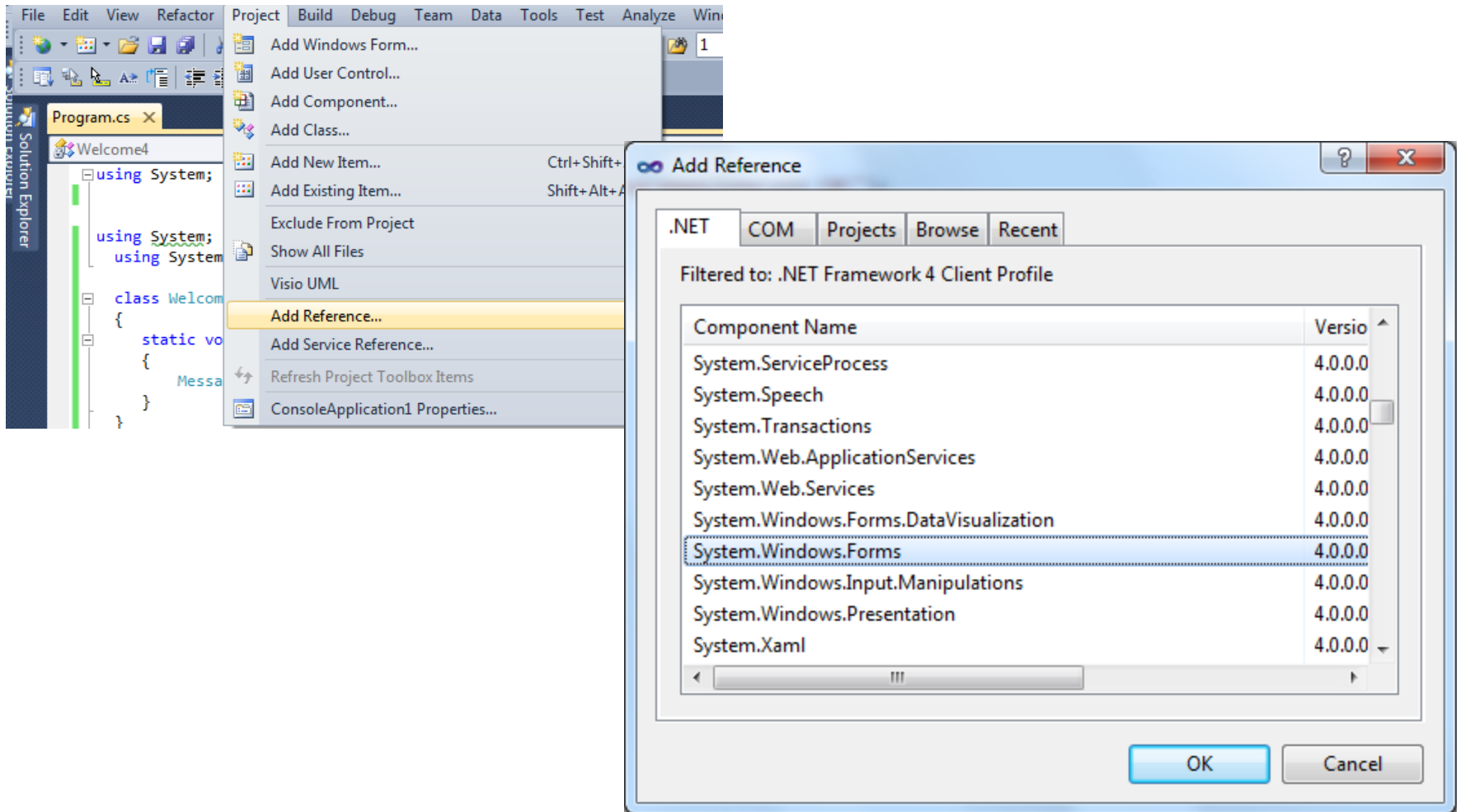
# Exception

```
try {  
    // codice che può portare ad un errore  
}  
catch (Exception Type [ variable ]) {  
    // codice che deve essere eseguito quando avviene l'errore  
}  
finally {  
    // codice da eseguire che avvenga o meno  
    // l'errore  
}
```

# Visualizzazione dell'Output

- ❑ **Con C# → Console applications**
  - Possiamo visualizzare solo testo
  - **Console.Write("... {0}", Sum);**
    - ❑ È simile alla printf con la differenza che i segnaposti non sono tipati ma solo enumerati ed indicati tra { } nella stringa sono ovviamente ammessi i caratteri speciali per come \0 \t ecc.
  - **Console.WriteLine("...");**
    - ❑ Stampa una riga di testo e va a capo

# Importazione di librerie



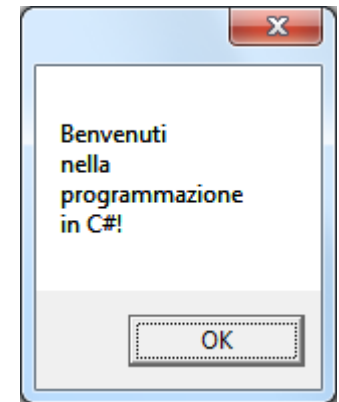
# Importazione di librerie

// Stampa di linee multiple in una DialoBox

```
using System;  
using System.Windows.Forms;
```

Il namespace System.Windows.Forms permette al programmatore di usare una classe MessageBox

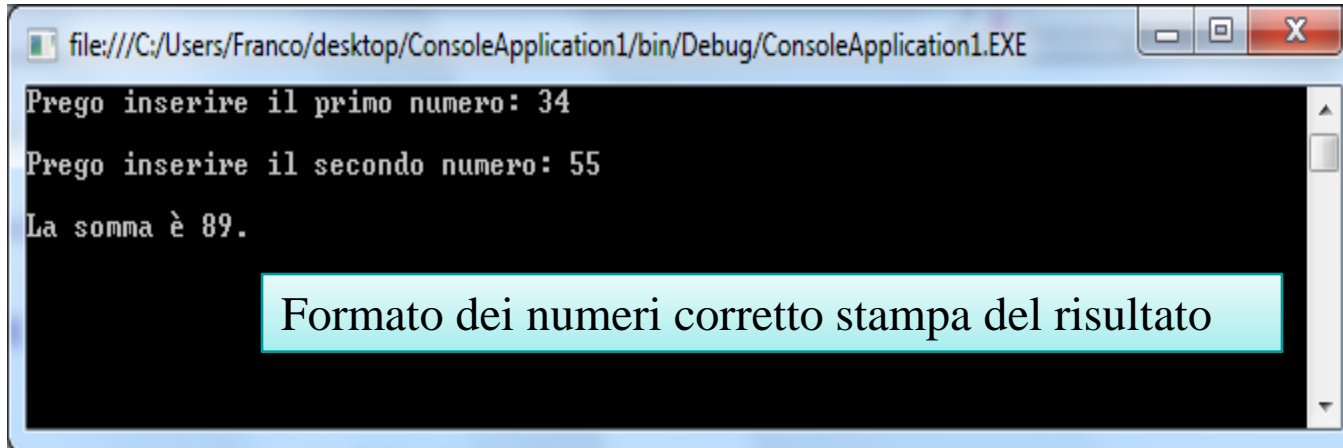
```
class Welcome  
{  
    static void Main( string[] args )  
    {  
        MessageBox.Show("Benvenuti\nnella\nProgrammazione\nC#!" );  
    }  
}
```



# Tipi di Dati Primitivi in C#

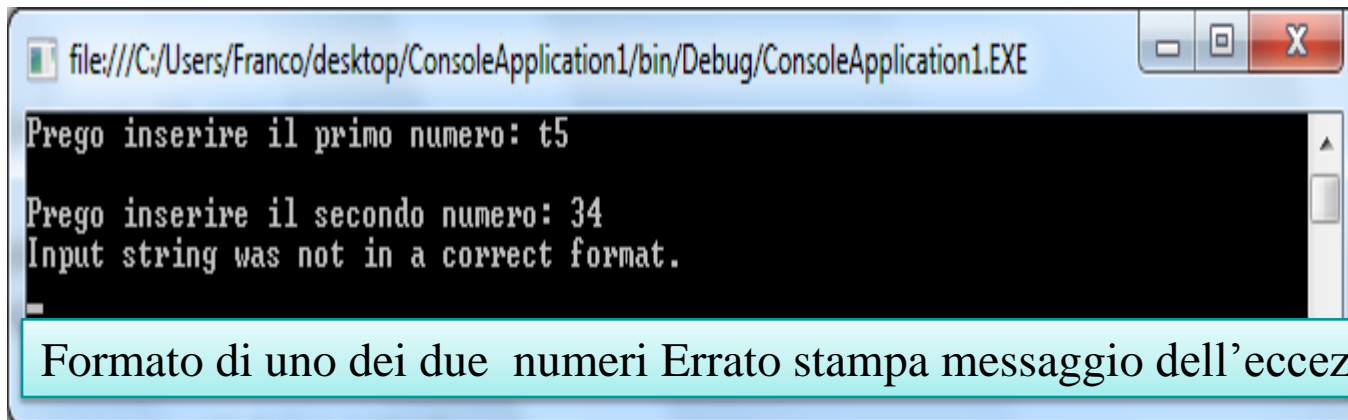
Nome breve	Classe .NET	Tipo	bit	Intervallo (bit)
byte	<a href="#">Byte</a>	Valore integer senza segno	8	Da 0 a 255
sbyte	<a href="#">SByte</a>	Valore integer con segno	8	Da -128 a 127
int	<a href="#">Int32</a>	Valore integer con segno	32	Da -2.147.483.648 a 2.147.483.647
uint	<a href="#">UInt32</a>	Valore integer senza segno	32	Da 0 a 4294967295
short	<a href="#">Int16</a>	Valore integer con segno	16	Da -32.768 a 32.767
ushort	<a href="#">UInt16</a>	Valore integer senza segno	16	Da 0 a 65535
long	<a href="#">Int64</a>	Valore integer con segno	64	Da -922337203685477508 a 922337203685477507
ulong	<a href="#">UInt64</a>	Valore integer senza segno	64	Da 0 a 18446744073709551615
float	<a href="#">Single</a>	Tipo a virgola mobile e precisione singola	32	Da -3,402823e38 a 3,402823e38
double	<a href="#">Double</a>	Tipo a virgola mobile e precisione doppia	64	Da -1,79769313486232e308 a 1,79769313486232e308
char	<a href="#">Char</a>	Singolo carattere Unicode	16	Simboli Unicode utilizzati nel testo
bool	<a href="#">Boolean</a>	Tipo booleano logico	8	True o false
object	<a href="#">Object</a>	Tipo di base per tutti gli altri tipi		
string	<a href="#">String</a>	Sequenza di caratteri		
decimal	<a href="#">Decimal</a>	Tipo frazionario o integrale esatto che può rappresentare numeri decimali con 29 cifre significative	128	Da $\pm 1.0 \times 10e-28$ a $\pm 7.9 \times 10e28$

# Esempio Somma di due numeri Output:



```
file:///C:/Users/Franco/desktop/ConsoleApplication1/bin/Debug/ConsoleApplication1.EXE
Prego inserire il primo numero: 34
Prego inserire il secondo numero: 55
La somma è 89.
```

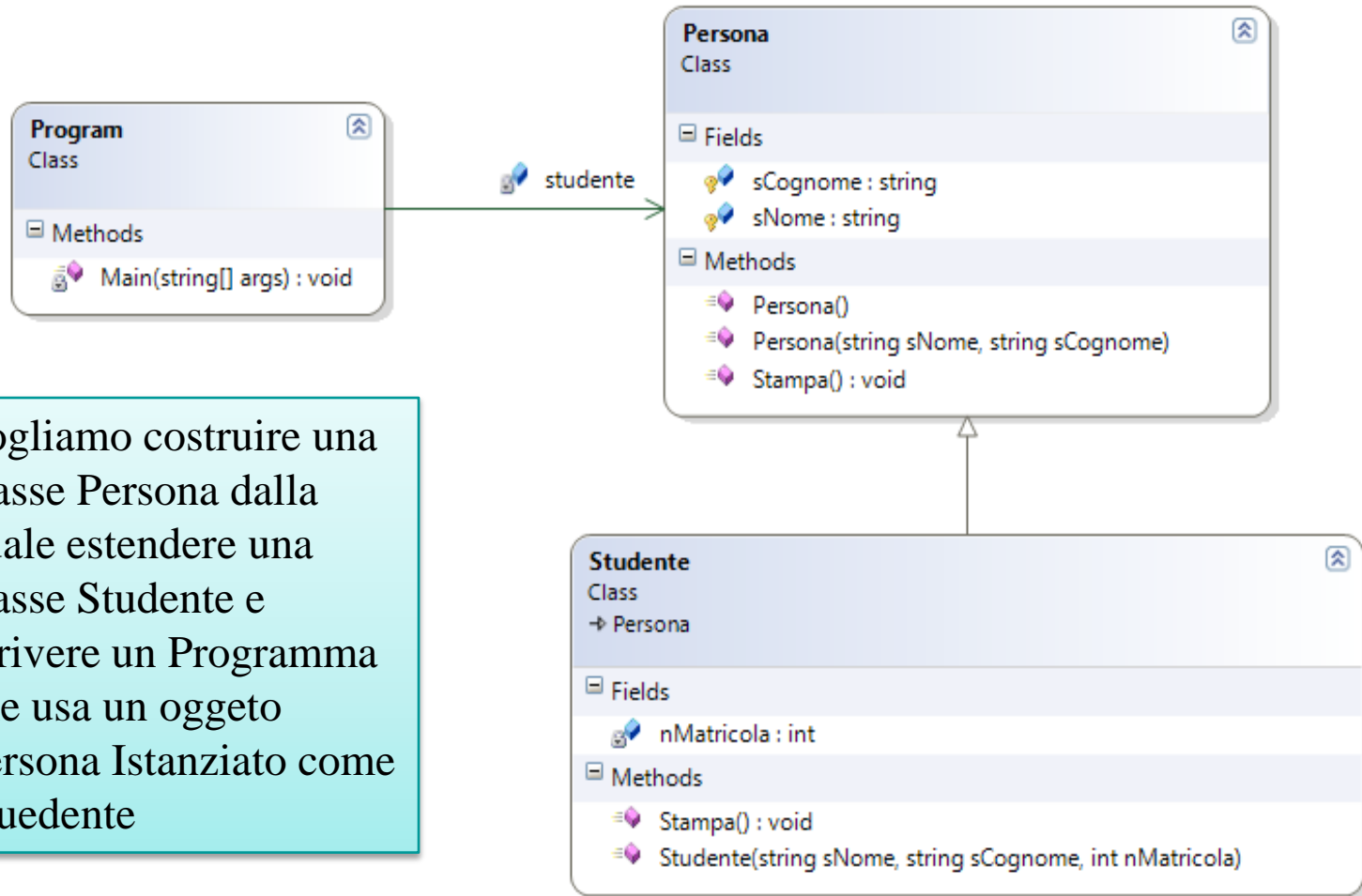
Formato dei numeri corretto stampa del risultato



```
file:///C:/Users/Franco/desktop/ConsoleApplication1/bin/Debug/ConsoleApplication1.EXE
Prego inserire il primo numero: t5
Prego inserire il secondo numero: 34
Input string was not in a correct format.
```

Formato di uno dei due numeri Errato stampa messaggio dell'eccezione

# Ereditarietà e Polimorfismo



Vogliamo costruire una classe Persona dalla quale estendere una classe Studente e scrivere un Programma che usa un oggetto Persona Istanziato come Studente

# Classe Persona

```
using System;
namespace classi_derivate
{
    public class Persona
    {
        protected string sNome;
        protected string sCognome;
        public Persona(string sNome, string sCognome)
        {
            this.sNome = sNome;
            this.sCognome = sCognome;
        }

        public Persona() { }

        public void Stampa()
        {
            Console.WriteLine(sNome + " " + sCognome);
        }
    }
}
```



# Classe Studente

```
using System;
namespace classi_derivate
{
    public class Studente : Persona
    {
        private Int32 nMatricola;

        public Studente(string sNome, string sCognome, Int32 nMatricola)
        {
            this.sNome = sNome;
            this.sCognome = sCognome;
            this.nMatricola = nMatricola;
        }
        public void Stampa()
        {
            Console.WriteLine(sNome + " " + sCognome + " " + nMatricola);
        }
    }
}
```

# Classe Programma

```
using System;
using System.Windows.Forms;
namespace classi_derivate
{
    class Program
    {
        static void Main(string[] args)
        {
            Persona studente;
            string nome, cognome;
            int matricola;
            Console.WriteLine("inserisci il nome dello studente");
            nome = Console.ReadLine();

            Console.WriteLine("inserisci il cognome dello studente");
            cognome = Console.ReadLine();
            Console.WriteLine("inserisci la matricola dello studente");
            try
            {
                matricola = Convert.ToInt32(Console.ReadLine());
                studente = new Studente(nome, cognome, matricola);
                studente.Stampa();
            }
            catch (FormatException fe) {
                Console.WriteLine(fe.Message.ToString());
                Console.WriteLine("la matricola deve contenere solo numeri");
                MessageBox.Show("errore");
            }
            Console.ReadKey();
        }
    }
}
```

# Esercitazione n°1

Creare le classi prima descritte nelle slide precedenti e dall'esecuzione del programma si dovrebbe avere un output simile al seguente:

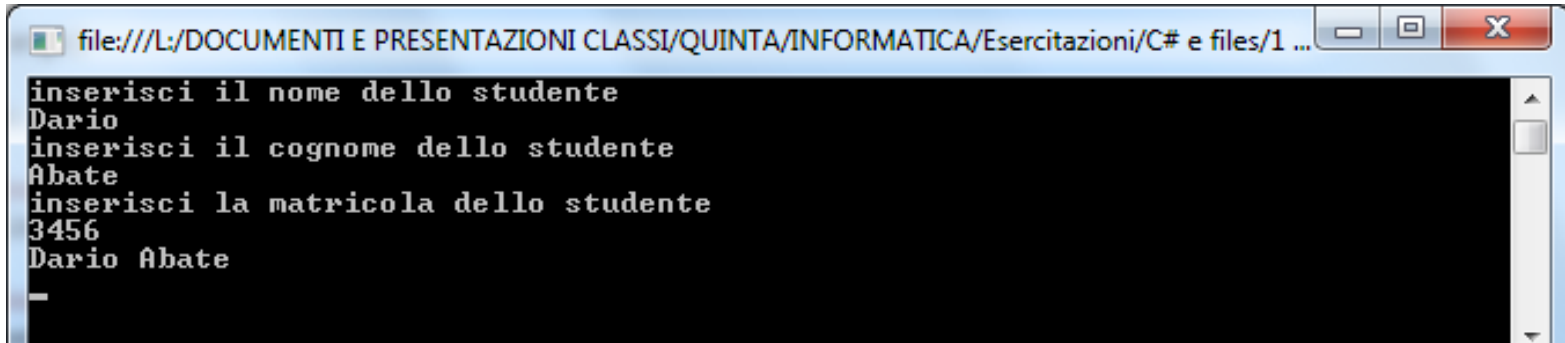
```
file:///L:/DOCUMENTI E PRESENTAZIONI CLASSI/QUINTA/INFORMATICA/Esercitazioni/C# e files/1 ...  
inserisci il nome dello studente  
Dario  
inserisci il cognome dello studente  
Abate  
inserisci la matricola dello studente  
3456  
Dario Abate  
-
```

```
file:///L:/DOCUMENTI E PRESENTAZIONI CLASSI/QUINTA/INFORMATICA/Esercitazioni/C# e files/1 ...  
inserisci il nome dello studente  
Dario  
inserisci il cognome dello studente  
Abate  
inserisci la matricola dello studente  
56r21  
Input string was not in a correct format.  
la matricola deve contenere solo numeri
```

errore  
OK

Matricola  
non corretta

# Esercitazione n°1



```
file:///L:/DOCUMENTI E PRESENTAZIONI CLASSI/QUINTA/INFORMATICA/Esercitazioni/C# e files/1 ...
inserisci il nome dello studente
Dario
inserisci il cognome dello studente
Abate
inserisci la matricola dello studente
3456
Dario Abate
-
```

Come Mai il Programma non stampa la matricola?

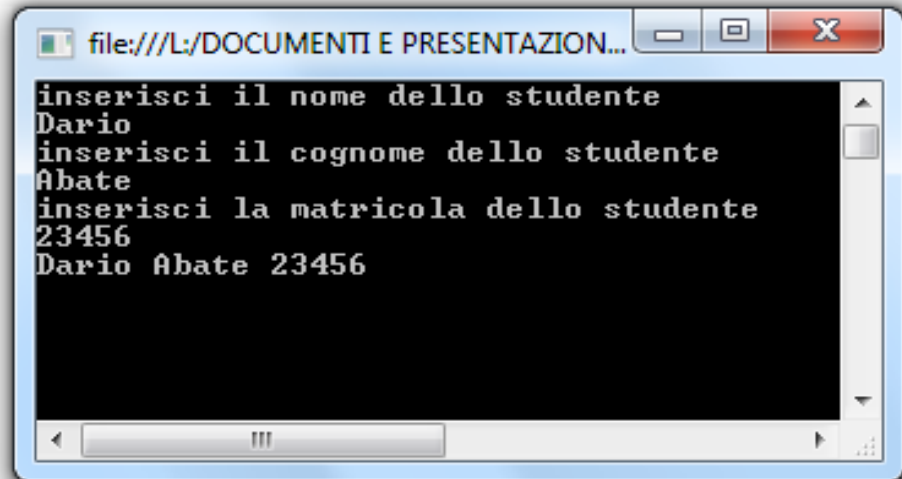
La risposta è semplice la nostra variabile studente è di tipo Persona. Quindi viene chiamato il metodo Stampa del padre.

Provate a Modificare studente In Studente verificate il risultato

# Esercitazione n°1

```
using System;
using System.Windows.Forms;
namespace classi_derivate
{
    class Program
    {
        static void Main(string[] args)
        {
            Studente studente;
            string nome, cognome;
            int matricola;
            Console.WriteLine("inserisci il nome dello studente");
            nome = Console.ReadLine();

            Console.WriteLine("inserisci il cognome dello studente");
            cognome = Console.ReadLine();
            Console.WriteLine("inserisci la matricola dello studente");
            try
            {
                matricola = Convert.ToInt32(Console.ReadLine());
                studente = new Studente(nome, cognome, matricola);
                studente.Stampa();
            }
        }
    }
}
```



```
file:///L:/DOCUMENTI E PRESENTAZION...
inserisci il nome dello studente
Dario
inserisci il cognome dello studente
Abate
inserisci la matricola dello studente
23456
Dario Abate 23456
```