

# Programmazione Orientata agli Oggetti in Linguaggio Java

## Ruoli e Responsabilità: C#

versione 1.1

Questo lavoro è concesso in uso secondo i termini di una licenza Creative Commons  
(vedi ultima pagina)

G. Mecca – Università della Basilicata – mecca@unibas.it



Ruoli e Responsabilità: C# >> Sommario



## Sommario

- In Sintesi
- Gli Esempi
- La Piattaforma .NET
  - ⇒ System.Collections.ArrayList
  - ⇒ System.Random
  - ⇒ Tipi Enumerati



## In Sintesi

- Nella programmazione C#
  - ⇒ valgono esattamente le stesse linee guida discusse a proposito di Java
- Interfaccia e implementazione
  - ⇒ incapsulamento
- Strati applicativi
  - ⇒ architettura di base



## Gli Esempi

>> mediaPesata  
>> indovina  
>> morraCinese

- Il codice degli esempi in C#
  - ⇒ molto simile per impostazione al codice degli esempi in Java
  - ⇒ naturalmente cambia l'utilizzo delle librerie
- I namespace
  - ⇒ Unibas.MediaPesata
  - ⇒ Unibas.Indovina
  - ⇒ Unibas.MorraCinese



## La Piattaforma .NET

- Il namespace System.Collections
  - ⇒ contiene le principali collezioni
  - ⇒ in particolare System.Collections.ArrayList
  - ⇒ implementa l'interfaccia System.Collection.IList
  - ⇒ e rappresenta liste di riferimenti a Object
  - ⇒ costruite con array e indicatore di riempimento
  - ⇒ senza limite sul numero di elementi



## La Piattaforma .NET

- I metodi
  - ⇒ Add(Object o)
  - ⇒ Insert(int i, Object o)
  - ⇒ Remove(int i)
  - ⇒ Clear()
  - ⇒ IndexOf(Object o)
  - ⇒ ...



## La Piattaforma .NET

- Due particolarità
  - ⇒ la proprietà Count e l'indicizzatore
- La proprietà Count
  - ⇒ consente di conoscere il numero di elementi
- L'indicizzatore
  - ⇒ consente di accedere agli elementi della lista come se fosse un array
  - ⇒ attraverso le parentesi []



```
namespace Unibas.Mediapesata {  
    public class Studente {  
        private System.Collections.ArrayList esami = new System.Collections.ArrayList();  
  
        public void AddEsame(Esame esame) {  
            this.esami.Add(esame);  
        }  
  
        public Esame GetEsame(int i) {  
            if (i < 0 || i >= this.esami.Count) {  
                throw new System.IndexOutOfRangeException("Esame inesistente");  
            }  
            return (Esame) this.esami[i];  
        }  
  
        public int NumeroEsami {  
            get {  
                return this.esami.Count;  
            }  
        }  
        ...  
    }  
}
```

l'indicizzatore associato  
alla classe ArrayList  
consente di accedere  
alla lista come se fosse  
un array  
in Java: (Esame)this.esami.get(i)



## La Piattaforma .NET

- **Attenzione**
  - ⇒ non esiste una versione della lista basata sulla rappresentazione collegata
- **Altre collezioni disponibili**
  - ⇒ `System.Collections.Stack`
  - ⇒ `System.Collections.Queue`
  - ⇒ `System.Collections.Hashtable`



## La Piattaforma .NET

- **La classe `System.Random`**
  - ⇒ generatore di numeri casuali
  - ⇒ inizializzato utilizzando l'orologio
  - ⇒ per generare la sequenza pseudo-casuale: metodo `int Next(int max)`

```
namespace Unibas.Indovina {  
  
    public class Partita {  
  
        private static System.Random generatore = new System.Random();  
        private string nome;  
        private int numeroDaIndovinare;  
        private int numeroDiTentativi;  
        private bool trovato;  
        private string suggerimento;  
  
        public Partita() {  
            this.numeroDaIndovinare = System.Math.Abs(Partita.generatore.Next(100) + 1);  
            this.trovato = false;  
            this.numeroDiTentativi = 0;  
            this.suggerimento = "Ho scelto un numero tra 1 e 100. Prova a indovinarlo.";  
        }  
    }  
}
```

## Tipi Enumerati

### ○ In Java

- ⇒ tradizionalmente i tipi enumerati sono rappresentati con costanti pub. di tipo intero
- ⇒ le enumerazioni sono disponibili da J2SE 5.0

### ○ In C#

- ⇒ è possibile utilizzare tipi enumerati fin dalla prima versione (funzionalità nativa)
- ⇒ anche in questo caso un tipo enumerato è una forma semplificata di classe



## Tipi Enumerati

### ○ Funzioni di un tipo enumerato in C#

- ⇒ definisce un nuovo tipo di dato
- ⇒ es: public enum Oggetto  
    { CARTA = 1, FORBICI, SASSO};
- ⇒ definisce un insieme di costanti statiche del tipo di dato con valore intero
- ⇒ es: Oggetto.CARTA = 1, Oggetto.FORBICI, Oggetto.SASSO



```
public enum Oggetto {CARTA = 1, FORBICI, SASSO};

public class Mano {

    private Oggetto giocataGiocatore;
    private Oggetto giocataComputer;

    public void Gioca(Oggetto giocataGiocatore) {
        if (!IsValid(jocataGiocatore)) {
            throw new System.ArgumentException("Giocata non valida");
        }
        this.GiocataGiocatore = giocataGiocatore;
        GeneraGiocataComputer();
    }

    public bool IsValid(Oggetto giocata) {
        return (giocata == Oggetto.CARTA || giocata == Oggetto.FORBICI ||
            giocata == Oggetto.SASSO);
    }

    private void GeneraGiocataComputer() {
        this.giocataComputer = (Oggetto)System.Math.Abs(generator.Next(3) + 1);
    }
}
```



## Tipi Enumerati

### ○ Nota

- ⇒ normalmente un tipo enumerato viene definito nello stesso file di una classe
- ⇒ ed appartiene allo stesso namespace

```
namespace Unibas.MorraCinese {
```

```
    public enum Oggetto {CARTA = 1, FORBICI, SASSO};  
    public enum EsitoMano {VINTADALCOMPUTER,  
                           VINTADALGIOCATORE, INPAREGGIO};  
    public class Mano {...}
```

```
}
```



## Riassumendo

- In Sintesi
- Gli Esempi
- La Piattaforma .NET
  - ⇒ System.Collections.ArrayList
  - ⇒ System.Random
  - ⇒ Tipi Enumerati





## Termini della Licenza

- This work is licensed under the Creative Commons Attribution-ShareAlike License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-sa/1.0/> or send a letter to Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.
- Questo lavoro viene concesso in uso secondo i termini della licenza "Attribution-ShareAlike" di Creative Commons. Per ottenere una copia della licenza, è possibile visitare <http://creativecommons.org/licenses/by-sa/1.0/> oppure inviare una lettera all'indirizzo Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.