

C# e il paradigma funzionale



Alberto Acerbis
Software Architect – Trainer
alberto.acerbis@intre.it



Ferdinando Santacroce
Architecture Modernization Enabler
ferdinando.santacroce@gmail.com

Sponsor



CONSORTIVM SERVICE S.r.l.



UNIPORDENONE

moltiplica il valore della formazione



La programmazione a oggetti organizza il codice attorno a oggetti con stato mutabile.

- OOP -> fai qualcosa



Nella programmazione funzionale, una funzione non cambia le cose: prende dei valori e ne restituisce altri

- OOP -> fai qualcosa
- FP -> dimmi che valore ottengo

Proprietà	OOP	FP
Gestione Dati (Stato)	I dati sono mutabili e encapsulati all'interno degli oggetti. L'aggiornamento dello stato è frequente.	I dati sono immutabili. Invece di modificare un dato esistente, in FP si crea una nuova struttura dati con le modifiche.
Elementi Base	Oggetti, classi, metodi, istanze.	Funzioni pure, funzioni di ordine superiore, immutabilità.
Approccio alla Logica	Imperativo. Descrive come eseguire i passaggi e cambiare lo stato per raggiungere un risultato.	Dichiarativo. Si concentra su cosa deve essere calcolato, non sui passaggi intermedi.
Gestione dei Comportamenti	Incapsulamento. Dati e metodi che operano su di essi sono uniti.	Funzioni pure. Non hanno side-effects e dipendono solo dagli input
Gestione delle Iterazioni	Usa cicli (for, while) per iterare sui dati.	Predilige la ricorsione e funzioni di ordine superiore (map, filter, reduce). Funzioni che accettano altre funzioni come parametri, o restituiscono funzioni come risultato.
Concorrenza	Può richiedere meccanismi di lock complessi a causa della mutabilità	Più semplice da gestire, poiché l'immutabilità evita stati condivisi modificabili.



Controller?



Minimal API?









Alberto Acerbis

Software Architect - Trainer

INTE3 25



Grazie!



ferdinando.santacroce@gmail.com



alberto.acerbis@gmail.com



<https://github.com/jesuswasrasta/car-sharp>



Ferdinando Santacroce
Architecture Modernization Enabler
ferdinando.santacroce@gmail.com