# Esercitazione di Fine Settimana – Week 2

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | Nome | Francesco |
|  |  | Cognome | Dell’Ascenza |
|  |  | Data | 02/04/2021 |

Leggete attentamente ogni domanda e argomentare quanto più possibile fornendo anche degli esempi.

1. Descrivere le modalità per ritornare più valori da un metodo in C#
2. Descrivere le due tipologie di casting tra tipi in C#
3. Quali sono gli utilizzi della keywork static?
4. Descrivere le differenze tra le interfacce utilizzabili per le collection
5. Cos’è un Generic?
6. Per far tornare più di un valore da un metodo con C# possiamo usare diversi modi, il modo classico è quello di farci tornare una classe che immagazzinasse i valori da tornare in un’istanza di questa, ma con le nuove versioni di C# possiamo adesso farci tornare più valori con le tuple. altro modo è tramite la keyword out, quando infatti specifico di che tipo sarà il valore di ritorno posso inserire tra i parametri un tipo aggiuntivo che sarà però non un parametro in input ma sarà un output aggiuntivo (introdotto dalla keyword out).
7. In C# si possono avere due tipi di casting, implicito ed esplicito, quello inplicito può avvenire solo da un tipo più particolare ad uno più generico ad esempio da int a double, perché in tal modo non viene persa informazione, al contrario quello esplicito viene richiesto quando si passa da un tipo più generico ad uno più particolare ad esempio da double a int.
8. La keyword static serve a definire proprietà o metodi statici di una classe, che esistono a prescindere delle varie istanze di questa, quindi sono appunto statici, se creo un metodo statico in una classe questa non ha bisogno di essere istanziata per richiamare il metodo, basta richiamarlo direttamente dalla classe. Allo stesso modo definire l’intera classe statica vuol dire assicurarsi che questa non verrà istanziata, dovrà solo essere richiamata per accedere ai metodi (anch’essi statici) contenuti in essa.
9. Le collections implementano interfacce di generics e grazie a queste definiscono (tramite l’implementazione di suddette interfacce) metodi aggiuntivi di queste che le permettono ad esempio con IEnumerable di poter iterare in esse con il foreach
10. Un generics è un modo per poter scrivere metodi e classi generali per qualsiasi type, si può ad esempio definire una classe generics che contiene liste di ogni type. Sono molto utili appunto per rendere il codice più generale e adattabile. Tuttavia il campo di accettazione dei type di un generic può essere limitato ad esempio specificando where T : IEsempio dove si restringe il campo dei types accettati a solo quelli che implementano I’interfaccia IEsempio. I generics non possono essere definiti null perché appunto accettando tutti i types potrebbero essere questi value types che non possono essere null.

**Esercitazione Pratica**

* Realizzare una classe Account per gestire un conto bancario, con le seguenti proprietà:
  + *Numero di Conto*
  + *Nome della Banca*
  + *Saldo*
  + *Data Ultima Operazione*
  + *Lista di Movimenti*
* Realizzare l’overload degli operatori + e – in modo che sia possibile aggiungere movimenti attivi e passivi alla lista dei movimenti (l’overload dovrà anche occuparsi di aggiornare il Saldo e la Data di Ultima Operazione)
* Realizzare un metodo Statement() che stampi i dati del conto, inclusa la lista dei movimenti
* Realizzare una gerarchia di classi per rappresentare movimenti bancari (Movement). Tutte le classi avranno le proprietà
  + *Importo*
  + *Data del Movimento*
  + Realizzare le classi che rappresentano:
    - CashMovement, con la proprietà aggiuntiva *Esecutore*
    - TransfertMovement, con le proprietà aggiuntive *Banca d’Origine* e *Banca Destinazione*
    - CreditCardMovement, con le proprietà aggiuntive *Tipo* (enum con i valori AMEX, VISA, MASTERCARD, OTHER) e *Numero di Carta*
  + Tutte le classi saranno dotate di costruttore che accetti tutti i parametri necessari per popolare le proprietà
  + Tutte le classi dovranno implementare la propria versione del metodo ToString() e visualizzare tutti i dati
* Realizzare una Console app che
  + Crei un nuovo Account
  + Permetta di inserire diversi tipi di Movimenti (input dall'utente)
  + Stampi i dati del conto e i movimenti