# Esercitazione di FineSettimana– Week 2

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | Nome | Valentyna |
|  |  | Cognome | Bogachova |
|  |  | 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#

Di solito un metodo può ritornare solo un valore, ma possiamo ampliarne un altro comportamento per ritornare più valori .Ritornare una classe o una struttura con tutti i valori necessari, utilizzo delle tuple e utilizzo della parola chiave out.

1. Descrivere le due tipologie di casting tra tipi in C#

Ci sono due tipologia di casting tra i tipi implicita ed esplicita .

Conversione implicita non è richiesta alcuna sintassi speciale perche la conversione avviene sempre con successo e i dati non vengano persi.

Conversione esplicita richiede una rappresentazione di cast , viene richiesto quando si potrebbero perdere informazioni nella conversione oppure per altri motivi la conversione non potrebbe riuscire.

1. Quali sono gli utilizzi della keywork static?

Viene utilizzata con i campi: un campo statico di una classe è un campo uguale per tutte le istanze di quella classe, viene assegnato con il nome della classe e non quello dell’istanza,metodoi ,un metodo statico è uguale per tutte le istanze, dovrà prendere in input l’istanza della classe su cui fare i calcoli con i classe,una classe statica non può essere istanziata con la parola new, i suoi campi e metodi possono essere utilizzati usando direttamente il nome della classe,i dati vengono salvati in una parte di memoria differente.

1. Descrivere le differenze tra le interfacce utilizzabili per le collection

Ogni interface può essere implementata con un determinato modo e avere varie tipologie di collection

1. Cos’è un Generic?

Costrutti di linguaggio che consentono di scrivere le classi, metodi ed interfacce indipendente dal tipo,al posto <T> si può mettere qualsiasi classi. I vantaggi del generics che possiamo riutilizzare il codice , meno codice da scrivere e da mantenere, un metodo/classe può essere utilizzato con tipi differenti.

**Esercitazione Pratica**

* Realizzare una classe Accountper 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