

Code kata

Record

Indice generale

| | | |
|----------|---|----------|
| 1 | Generare un vettore di record..... | 3 |
| 2 | Modificare i dati di un vettore..... | 4 |
| 3 | Filtrare i dati di un vettore/lista..... | 5 |
| 4 | Ricerca di un elemento in un vettore..... | 6 |
| 5 | Creare un elenco di record da una matrice di stringhe..... | 7 |
| 6 | Memorizzare le voci di un dizionario..... | 8 |

1 Generare un vettore di record

Problema

Dato il tipo: `Atleta`, generare un vettore di atleti. (Utilizzare dei dati a scelta.)

```
class Atleta
{
    public string Nominativo;
    public double Altezza;
    public double Peso;
}
```

Esempio:

```
Atleta[] atleti = GetAtleti(); // ->
    {Nominativo:"Rossi, Andrea"; Altezza:1,91; Peso:92}
    {Nominativo:"Verdi, Sonia"; Altezza:1,70; Peso:57}
    {Nominativo:"Bianchi, Filippo"; Altezza:1,77; Peso:68}
```

2 Modificare i dati di un vettore

Problema

Dato il tipo: `Ordine`:

```
class Ordine
{
    public string Descrizione;
    public double CostoUnitario;
    public int Quantità;
}
```

e dato un vettore di ordini, diminuire del 10% il costo unitario. Modificare direttamente gli ordini del vettore.

Variazioni

1. Modifica il procedimento in modo da elaborare una lista di ordini invece di un vettore.
2. Modifica il procedimento in modo che sia possibile specificare la % di riduzione del costo.
3. Modifica il procedimento in modo che restituisca un nuovo vettore contenente una copia degli ordini originali con il costo unitario diminuito.

3 Filtrare i dati di un vettore/lista

Problema

Data una lista di atleti, restituire una nuova lista contenente solo quelli più alti di 1,80.

Esempio

```
List<Atleta> atleti = GetAtleti(); //  
    {Nominativo:"Rossi, Andrea"; Altezza:1,91; Peso:92}  
    {Nominativo:"Verdi, Sonia"; Altezza:1,70; Peso:57}  
    {Nominativo:"Bianchi, Filippo"; Altezza:1,77; Peso:68}  
  
//-> {Nominativo:"Rossi, Andrea"; Altezza:1,91; Peso:92}
```

Variazioni

1. Generalizza il procedimento per un'altezza qualsiasi.
2. Supponi che gli atleti siano in un vettore. Modifica il procedimento in modo che restituisca un vettore degli atleti più alti del valore specificato.

Requisiti (necessari / utili) per la soluzione

Trasformare una lista in un vettore; metodo `ToArray()`.

4 Ricerca di un elemento in un vettore

Problema

Dato un vettore di atleti e dato un nominativo, restituire la posizione dell'atleta corrispondente nel vettore. Restituire -1 se non esiste un atleta con quel nominativo.

Esempio:

```
Atleta[] atleti = GetAtleti();
    {Nominativo:"Rossi, Andrea"; Altezza:1,91; Peso:92}
    {Nominativo:"Verdi, Sonia"; Altezza:1,70; Peso:57}
    {Nominativo:"Bianchi, Filippo"; Altezza:1,77; Peso:68}
string nominativo = "Verdi, Sonia";
// -> 1
```

Variazioni

1. Fornisci versioni del procedimento con il ciclo `for` e `foreach`.
2. Modifica il procedimento in modo che restituisca `true` se esiste, `false` se non esiste.

5 Creare un elenco di record da una matrice di stringhe

Problema

Sia dato una matrice di stringhe contenente le informazioni di una fattura. Creare un elenco di record che memorizza le stesse informazioni in forma tipizzata (mediante dei record):

```
string[,] fattura =  
{  
    // Cod. Prod.   descrizione      qta    prezzo unitario  
    {"CO1",        "COMPUTER HP",    "1",    "1000"},  
    {"MT1",        "MONITOR DELL",  "2",    "500"},  
    {"CD1",        "CD-ROM PHILIPS", "5",    "40"},  
    {"TA1",        "TASTIERA TRUST", "10",   "10"}  
};  
  
"voce = descrizione"
```

Creare un vettore di record del seguente tipo:

```
class Ordine  
{  
    public string CodiceProdotto;  
    public string Descrizione;  
    public int Quantità;  
    public double PrezzoUnitario;  
}
```

6 Memorizzare le voci di un dizionario

Problema

Sia dato un vettore di stringhe che rappresenta un dizionario. Il contenuto di ogni stringa è nella forma:

“voce = descrizione”

Creare un vettore di record del seguente tipo:

```
class VoceDizionario
{
    public string Voce;
    public string Descrizione;
}
```

Esempio:

```
string[] dizionario =
{
    "IDE = Ambiente di sviluppo integrato",
    "Debugging = Eliminazione degli errori",
    "... = ..."
};
//->
// {Voce:"IDE"; Descrizione:"Ambiente di sviluppo integrato"}
// {Voce:"Debugging"; Descrizione:"Eliminazione degli errori"}
// {Voce:"..."; Descrizione:"..."}
```

Variazioni

Considera la possibilità che alcune stringhe del vettore siano vuote/nulle; in quel caso, scartale.

Requisiti (necessari / utili) per la soluzione

Dividere una stringa usando un determinato carattere come separatore; metodo **Split()**:

```
string testo = "Altezza: 1,98";
string[] dati = testo.Split(':'); //-> "Altezza", " 1,98"
```