

## Il programmatore che c'è in noi - Lezione 4 - Variabili

Le variabili, in un programma, sono i "contenitori" dentro cui memorizzare dei valori, che possono cambiare (per questo si chiamano variabili) durante l'esecuzione del programma stesso.

Quindi per semplificare il concetto di variabile, immaginiamo di avere delle scatole (come avevamo già fatto parlando dei tipi di dati), ma questa volta diamo un nome alle scatole, in modo da poterle facilmente individuare.

Le azioni che posso fare, usando queste scatole, sono sostanzialmente di "vedere" cosa contiene una scatola oppure "mettere" qualcosa nella nostra scatola.

Ovviamente, riferendomi alle lezioni sui tipi di dato, una scatola è "personalizzata" in base al tipo di dato che essa conterrà.

La sintassi (ossia il modo in cui devo scrivere un'istruzione in un linguaggio di programmazione) per poter dichiarare una variabile in linguaggio C / C++ e ObjectiveC è la seguente

Tipo di dato	Nome della variabile	Valore iniziale
--------------	----------------------	-----------------

Vediamo degli esempi:

Se nel mio programma dovrò gestire dei dati di tipo intero esempio l'età di due persone (mio padre e mia madre) avrò la necessità di usare due VARIABILI (ossia due scatole, dentro cui mettere i valori relativi agli anni dei miei genitori).

Per dichiarare (ossia, creare, far nascere, all'interno del programma una variabile) mi pongo alcune semplici domande:

- 1) **che tipo di dato deve contenere la variabile ?**

La risposta a tale domanda, determina il TIPO DI DATO da usare nella sintassi per la dichiarazione della variabile stessa

- 2) **quale è l'informazione che sarà gestita dalla variabile ?**

La risposta a tale domanda, mi aiuta a scegliere il NOME DELLA VARIABILE.

### 3) ha un valore iniziale ?

In questo caso, la mia risposta è sempre SI, il valore iniziale io lo metto sempre, anche se vale 0.

Nel esempio, il tipo di dato che deve contenere l'età, potrebbe essere un int, un unsigned short, un char o un double.

Ovviamente, la scelta migliore deve essere fatta tenendo in considerazione, quando possibile, l'ottimizzazione dell'utilizzo della memoria RAM del nostro Mac.

Scelgo, unsigned char

Per il nome, decido nAnniPapa e nAnniMamma

#### NOTA:

Non esiste un criterio universalmente riconosciuto per scrivere il nome della variabile.

Come regole di base, **non usate** simboli di punteggiatura, non scrivete tutto in maiuscolo, non iniziate un nome di variabile con un numero, non lasciate spazi tra le parti componenti il nome.

Comunque, se il nome che avete scelto non è accettabile dal "compilatore" (=programma che elabora le nostre istruzioni di programma scritte in un certo linguaggio di programmazione, e le trasforma in istruzioni per il microprocessore del nostro Mac) sarà segnalato un errore nel momento in cui cercherete di mandare in esecuzione il programma (per come fare a compilare e mandare in esecuzione un vostro programma, scaricate il tutorial su XCODE della puntata precedente).

Cercate sempre di usare un nome di variabile che sia il più descrittivo possibile (evitando di scrivere un poema...) questo è di FONDAMENTALE importanza, perché quando rimetterete le mani su un vostro programma a distanza di tempo, sarete in grado di risalire alle funzionalità del programma ed allo scopo delle variabili. La stessa cosa vale se dovrete mettere le mani su programmi fatti da altri e viceversa.

Vediamo quindi come scriveremo nel nostro programma le due variabili di cui abbiamo parlato

Tipo Dato	Nome variabile	Valore Iniziale
unsigned char	nAnniPapa	0
unsigned char	nAnniMamma	0

## ASSEGNAZIONE DI UN VALORE AD UNA VARIABILE

Per mettere qualcosa nella scatola si usa l'operatore di assegnazione indicato dal simbolo =

Quindi per assegnare un valore ad una variabile si usa l'operatore =

```
unsigned char nAnniPapa=45; //dichiaro la variabile e gli
assegno un valore
unsigned char nAnniMamma=38; //dichiaro la variabile e gli
assegno un valore
```

Esaminiamo questa riga di codice

```
unsigned char nAnniPapa=45; //dichiaro la variabile e gli
assegno un valore
```

Va letta nel modo seguente

"assegnamo il valore numerico intero pari a 45 alla variabile di tipo unsigned char che si chiama nAnniPapa"

E scomponiamola nei vari elementi.

`unsigned char`

indica il tipo di dato della nostra variabile

`nAnniPapa`

indica il nome della nostra variabile

`=`

indica l'operazione di assegnamento di un valore ad una variabile

`45`

indica il valore che intendo assegnare alla variabile

`;`

indica la fine di una istruzione in linguaggio C/C++/ObjectiveC

`//`

indica l'inizio di un commento. Da questo punto in poi tutto quello che e' scritto sulla riga e' da considerarsi un commento

Nella definizione iniziale di variabile ho detto che il valore di una variabile puo' cambiare durante l'esecuzione di un programma, semplicemente basta assegnare ogni volta che si vuole un nuovo valore usando l'operatore = (assegnazione).

Esempio

```
//Dichiarazione delle variabili
unsigned char nAnniPapa=45; //Anni Papa'
unsigned char nAnniMamma=38; //Anni Mamma

//Ora voglio assegnare nuovi valori
nAnniPapa = 52; //Assegno il valore 52 alla variabile
nAnniPapa
nAnniMamma = 45;
```

Posso anche assegnare il valore di una variabile ad un'altra  
In questo modo

```
//Supponiamo che papa' e mamma abbiano la stessa eta'
nAnniPapa= nAnniMamma;
```

Posso assegnare il risultato di un calcolo

```
nAnniPapa = 2006 - 1966;
```

oppure

```
nAnniPapa = nAnniMamma+5;
```

Per i piu' curiosi...

Tecnicamente parlando, la variabile che si trova alla SINISTRA del simbolo = (assegnazione) si chiama LVALUE, ricordatevi questo nome, in alcuni casi il compilatore vi potrebbe segnalare questo errore: invalid lvalue in assignment , che significa appunto che c'e' un errore nella parte sinistra di un operatore di assegnazione, l'operazione non puo' essere fatta.

Esempio di tale errore

```
7=5; //7 Non e' un lvalue e' un valore costante e quindi non
puo' essere modificato
```

Vi lascio sperimentare la dichiarazione delle variabili all'interno di XCODE, ho preparato a tal proposito questo semplice progetto.

Esempi di dichiarazioni e di assegnazione di variabili

```
double nTotCostoCassetta=0.0;  
int nNumeroDiMele=0;  
double nCostoMela=1.5;
```

```
//Assegnazione dati  
nNumeroDiMele=100;
```

```
//Calcolo costo totale e assegno il risultato alla variabile  
nTotCostoCassetta = nNumeroDiMele * nCostoMela;
```

Saluti.