

14

gen 2009

[Stampa articolo](#) 48 Commenti [[Modifica](#)]

Corso interattivo di programmazione per iPhone: I Lezione

[Corso Inglese Online](#)

Facile, Rapido e Divertente Corso a distanza.
Provalo Gratis!

Annunci Google

Con questa prima lezione inizia oggi il "Corso Interattivo di programmazione per iPhone" tenuto dallo sviluppatore italiano Alvisè Susmel e che ci guiderà, di lezione in lezione, dalle basi fino alla creazione di vere e proprie applicazioni per iPhone. Buon corso a tutti.



Introduzione

In questa serie di lezioni cercherò di spiegarvi passo passo come cominciare a programmare per iPhone. Prima di parlare subito su **come** fare la prima applicazione per iPhone, sarà necessario studiare abbastanza in dettaglio il linguaggio su cui si basa Cocoa: Objective-C 2.0. Una solida conoscenza di questo linguaggio vi consentirà di poter realizzare idee complesse in modo semplice. Quello che cercherò di insegnarvi sarà soprattutto il **metodo ingegneristico** e l'**approccio** con cui risolvere un certo tipo di problemi.

La prima parte del corso sarà quindi orientata più ad un aspetto teorico:

- Basi di programmazione ad oggetti
- Objective-C 2.0
- Gestione della memoria
- Alcuni oggetti Cocoa

Il vantaggio che avrete da questo tipo di trattazione teorica, sarà che non solo saprete programmare per iPhone, ma con uno sforzo pari a zero potrete programmare anche per il vostro Mac.

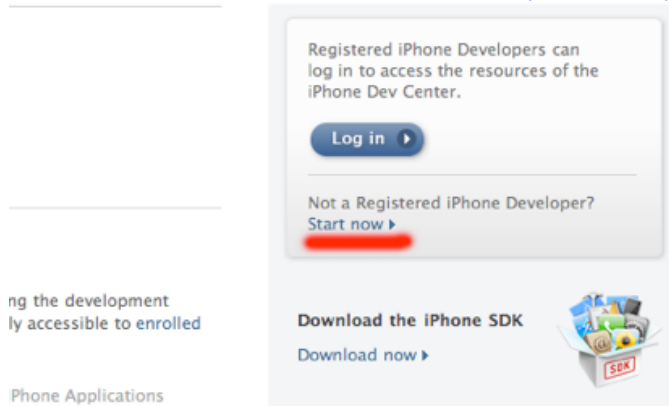
Questa prima parte è d' introduzione. Spiegherò come scaricare, installare l' iPhone SDK e un po' com'è strutturato.

Download dell' SDK

Come prima cosa dobbiamo scaricare l' iPhone SDK (Software Development Kit). Come dice

l'acronimo questo è il pacchetto che ci permette di sviluppare applicazioni per iPhone. All'interno troverete tutto: editor, compilatore (vi spiegherò che cos'è), programmi di debug, programmi per disegnare l'interfaccia grafica, simulatore iPhone, documentazione, etc.

Prendiamo Safari e andiamo nel sito <http://developer.apple.com/iphone>



Per scaricare l'SDK sarà necessario registrarsi (gratuitamente). Dovrete cliccare su "Start now", scritta che ho sottolineato in rosso nell'immagine.

Per chi non avesse già un account (un ID Apple tipo @mac.com o @me.com) dovrà registrarsi cliccando su "Create Apple ID".

Need to create an Apple ID?

An Apple ID is required to begin the registration process.

[Create Apple ID](#)

Dovrete compilare il classico form d'iscrizione, con password, nome cognome etc.

Create an Apple ID

You are required to create an Apple ID and password to register as an iPhone Developer. The Apple ID and password associated with an ADC membership, iTunes or Apple Store account.

Apple ID * Example: steve@mac.com

Account Password *

Re-enter Password *

Please make up a question and answer to help us verify your identity if you forget your password.

Security Question *

Answer *

BirthDay * Select Month Select Day

Personal Profile

You must provide a physical address. We cannot accept PO Boxes, APO or FPO addresses.

First *

Middle Initial

Una volta ottenuto l'Apple ID potrete fare il login su <http://developer.apple.com/iphone> e potrete scaricare non solo l'SDK ma anche la documentazione ufficiale (che vi consiglio caldamente), esempi di applicazioni, video.

Tornando nella pagina: <http://developer.apple.com/iphone>

Cliccate su "Login" e accedete alla pagina con i permessi per poter scaricare i vari contenuti.

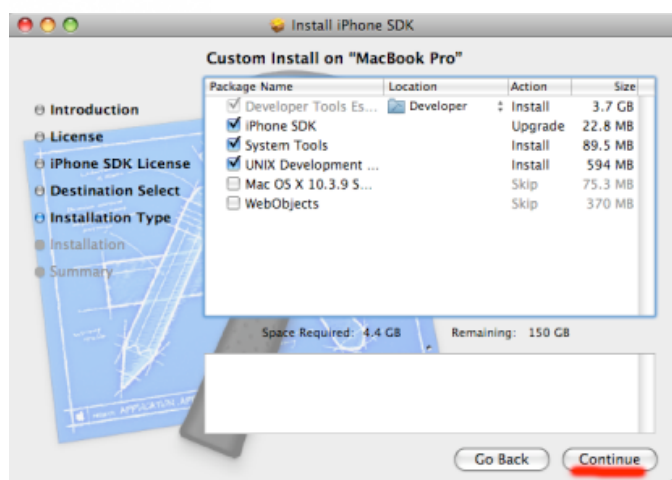
Cliccando su "iPhone SDK for iPhone OS 2.2" comincerete il download dell'SDK.



Ogni volta che il firmware dell' iPhone viene aggiornato dovremo scaricare la nuova versione dell' SDK. Questo perchè ogni volta che il sistema operativo viene aggiornato, anche le funzionalità dell' iPhone cambiano ed aggiornando l' SDK potremo usufruire di queste nuove funzionalità.

Installazione dell' SDK

Passiamo all' installazione dell' SDK. Avviate "iPhone SDK" come nell' immagine.

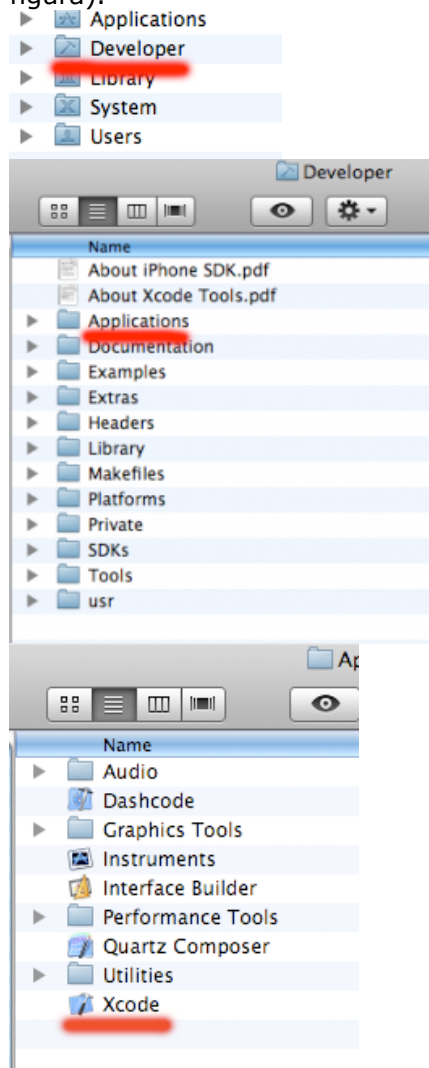


Lasciando invariate le selezioni premete "Continue" installando così tutto il necessario.

Directory /Developer

Alla fine dell' installazione troverete una directory "Developer" nella vostra partizione (come nella

figura).

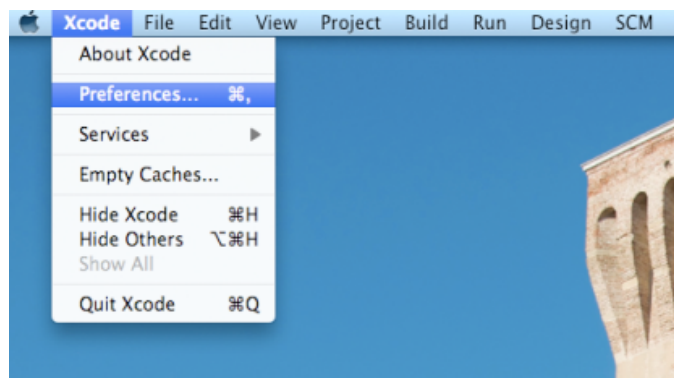


Dentro la directory /Developer trovate tutto. Per ora ci interesseremo solo di Xcode. Xcode è il programma che ci permette di accedere a tutte le funzionalità dell' SDK.

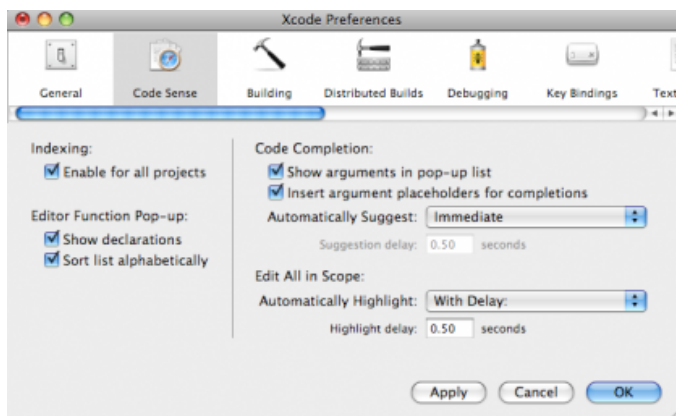
Breve introduzione a Xcode

Avere un editor ben impostato significa lavorare in un ambiente piacevole e quindi lavorare più velocemente e meglio. Una delle prime cose da configurare, quindi, è l' editor.

Dopo aver avviato Xcode, che si trova in /Developer/Applications/Xcode, andate in preferenze.

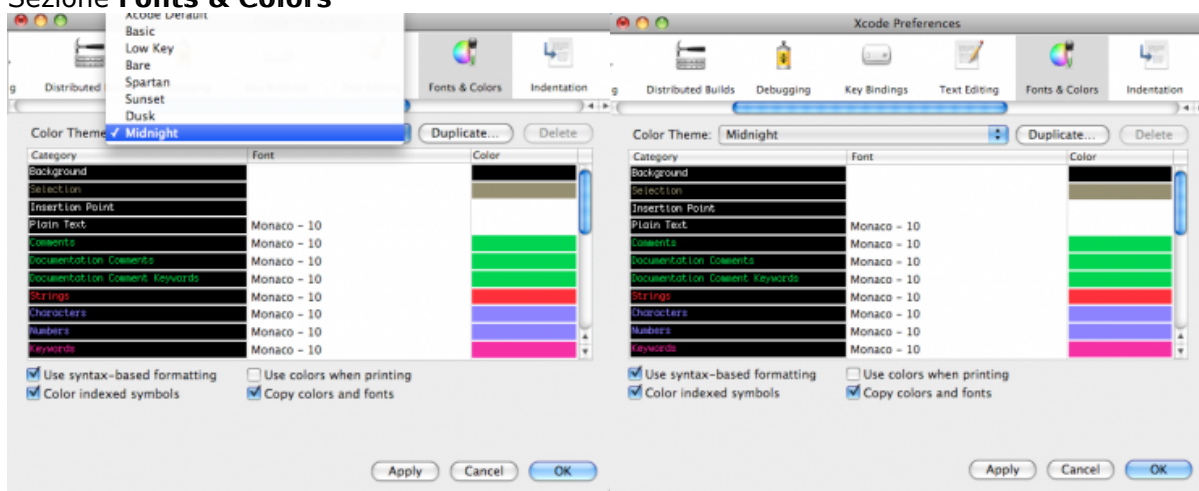


Sezione **Code Sense**



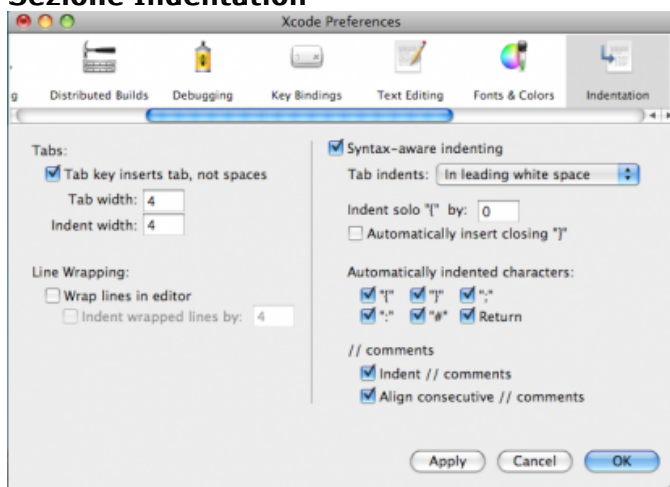
Come si vede dall' immagine in Code sense ci sono diverse opzioni da selezionare o delezionare. Il "pop-up" vi permette di vedere i vari metodi di una classe quando state scrivendo del codice. I settaggi dell' immagine sono quelli che utilizzo solitamente.

Sezione Fonts & Colors



In questa sezione potete cambiare i colori ed i font. Io utilizzo un tema predefinito "Midnight" che imposta il colore nero per lo sfondo e diversi colori per le varie parole che hanno un certo significato all' interno del codice.

Sezione Indentation



Nella scrittura di codice di qualsiasi tipo vengono usate molto le indentazioni.

```
int funzione(int argomento) {
    char *variabile;
    /// indentazione
}
```

In questa immagine ho scritto l'inizio della dichiarazione di una funzione in C.

La prima riga è l'inizio della dichiarazione di una funzione (vedremo in dettaglio avanti).

Vediamo che la scritta "char *variabile;" è più a destra, nel caso specifico c'è un'indentazione con un tab equivalente a 4 caratteri.

La funzione che dichiaro è la funzione **somma** che prende due numeri interi come argomento e ritorna la somma (questo è un esempio, nelle prossime lezioni tratterò in dettaglio la dichiarazione di funzioni, di variabili, di classi).

Funzione tutta in una riga

```
long int somma(int a, int b){long int c;c=a+b;return c;}
```

Anche se la funzione è semplice, il codice non è ordinato, non è facilmente leggibile.

Dobbiamo pensare che anche altre persone (se lavoriamo in team o se siamo in un progetto opensource) leggeranno il nostro codice. Potrebbe anche essere che in futuro noi stessi dovremo rileggere il nostro codice e preferiremo sicuramente una scrittura leggibile. Questo discorso fa parte di un discorso molto più ampio che riguarda lo stile dello scrivere codice. La stessa

```
long int somma(int a, int b){
    long int c;
    c=a+b;
    return c;
}
```

funzione scritta sfruttando le indentazioni

Dal punto di vista pratico le due immagini sono la stessa identica cosa.

Sempre nella sezione Indentation delle preferenze del nostro Xcode, troviamo l'opzione "Automatically insert closing '}' "

In C/C++, Java, Objective-C etc.. si fa un largo uso delle parentesi graffe. Queste servono per definire blocchi di codice. Quando ne apriamo una ci sarà sicuramente un momento in cui dovremo chiuderla, se selezionate questa opzione l'editor la chiuderà subito per voi. Se selezionate le opzioni "Automatically indented characters", dite all'editor di utilizzare l'indentazione quando utilizzate quei caratteri.

Ad esempio nel caso "{" il codice che scrivete verrà spostato di 4 spazi a destra, dopo aver inserito "}" ritornerete a scrivere il codice a sinistra di 4 spazi.

```
prima
{
    testo indentato
}
dopo|
```

L'opzione di AutoIndentazione è una cosa che vi consiglio, aumenta molto la velocità di scrittura del codice.

Conclusione della prima lezione

Oggi abbiamo visto in dettaglio come registrarsi, scaricare ed installare l'iPhone SDK. Abbiamo visto una piccola parte delle preferenze di Xcode che riguardano l'editor.

Nella prossima lezione inizieremo a creare un progetto di esempio, vi spiegherò che file bisogna

utilizzare e perchè, proveremo anche il simulatore iPhone. Utilizzeremo in modo totalmente elementare il programma Interface Builder che permette di fare interfacce grafiche per MacOS X e per iPhone in modo semplice e veloce. Farò una piccola introduzione all' Objective-C, se siete curiosi e volete già iniziare a leggervi qualche cosa su l' Objective-C, ora che siete registrati, potete guardare questo link:

<http://developer.apple.com/iphone/gettingstarted/docs/objectivecprimer.action>

Tags: [Applicazioni](#), [corso](#), [developer](#), [guida](#), [iphone](#), [programmare](#), [sdk](#), [Tutorial](#)

[Condividi](#)

Scritto da **Alvise**.

48 Commenti a "Corso interattivo di programmazione per iPhone: I Lezione"

[Scrivi un commento](#)



iMatt

[[Modifica](#)] 14 gennaio 2009 alle 12:40

Wow... Penso che ci proverò 😊



Riccardo

[[Modifica](#)] 14 gennaio 2009 alle 12:54

Io ho una domandina.....io sono un programmatore ma lato Microsoft.

L'SDK esiste anche per Windows XP?

grazie e complimenti per questo corso! Penso sia utilissimo



[Nico](#)

[[Modifica](#)] 14 gennaio 2009 alle 13:01

@riccardo: no, l'sdk esiste solo per mac os x /n.



gabriele