**Introduzione a**

**Visual Studio Code**

**Sergio Frasca**

# **Introduzione**

(da Wikipedia)

“Visual Studio Code è un editor di codice sorgente sviluppato da Microsoft per Windows, Linux e macOS. Include il supporto per debugging, un controllo per Git integrato, syntax highlighting, IntelliSense, snippet e refactoring del codice. Sono personalizzabili il tema dell'editor, le scorciatoie da tastiera e le preferenze. È un software libero e gratuito, anche se la versione ufficiale è sotto una licenza proprietaria.”

Si può scaricare da [Visual Studio Code - Code Editing. Redefined](https://code.visualstudio.com/) ed installare.

Può essere usato per vari linguaggi di programmazione (C, C++, C#, CUDA C++, HTML, Java, JavaScript, Latex, Python, R, Visual Basic ed altri) ed è attualmente il tool di sviluppo software più diffuso.

Oltre alla documentazione ufficiale, esistono vari libri che lo descrivono.

In questo documento presento un uso di VSC, non l’unico, ma quello che per ora ho trovato più comodo. Se qualcuno ha qualche dritta da propormi, gliene sono grato.

# **Setup**

La finestra che si apre può essere sistemata in vario modo. Questo è un esempio dato dall’help:

Immagine che contiene testo, schermata, monitor, nero

Descrizione generata automaticamente

Distinguiamo in questo setup:

* una Activity Bar
* una Side Bar
* due editor di file affiancati
* un pannello che in questo caso contiene un terminale, da cui si può lanciare Python o eseguire comandi di sistema
* una Status Bar

Ed ecco uno dei miei normali modi di lavorare:

Immagine che contiene testo, monitor, schermata, nero

Descrizione generata automaticamente

Si nota che l’activity bar è più “ricca” (sono state aggiunte alcune estensioni).

La side bar contiene

* gli editor aperti, da cui si possono salvare i file modificati o creati
* i file nel folder settato (SnagPy), da cui si possono aprire in editor, creare nuovi file o folder
* l’outline, cioè classi, funzioni e variabili del file in focus (in questo caso BASIC.py),
* una timeline dei salvataggi dei file aggiornati eseguiti: cliccando su un elemento, compare la vecchia e la nuova versione del file.

Nell’”editor groups” si vedono vari tab per vari file, ma uno solo è visibile (“in focus”); se ne possono rendere visibili anche più.

Dal nome del file aperto in editor si possono individuare e accedere direttamente le funzioni e le classi, e da queste le definizioni delle variabili, similmente a come accade in outline, ma in modo più comodo.

Nel pannello inferiore si possono visualizzare varie consolle: Problems, Output, Debug e Terminal. In questo caso è aperto un terminal, da cui è stato lanciato Python. Se si apre Python da questo terminale, possono essere mandate in esecuzione varie righe di comandi, copiati per esempio da un file in editor: alla partenza per esempio possono mandarsi dei comandi copiati da starting.py o snippets.py e incollate nella shell di Python aperta nel terminale: compare la seguente finestra

Immagine che contiene testo

Descrizione generata automaticamente

e spingendo Paste i comandi vanno in esecuzione (meno l’ultimo, per cui occorre un ulteriore return).

Nella Status Bar ci sono varie informazioni utili, e la possibilità di cambiare il linguaggio.

Vediamo ora qualche settaggio.

**La Menu Bar:** 

* **File** permette di aprire o creare file, aprire un folder e varie altre operazioni
* **Edit** permette di fare undo e redo, toggle # ed altro
* **Selection** varie operazioni su blocchi di codice selezionati
* **View** vari settaggi della finestra
* **Go** per cercare definizioni, assegnazioni di variabili, file o altro
* **Run** gestione del debugging
* **Terminal** crea nuovi terminal o manda in esecuzione file o pezzi selezionati
* **Help** help molto ricco (molto di più di queste note introduttive), con video tutorial, completa documentazione e altro

Infine, sulla destra ci sono dei pulsanti per far apparire o sparire pezzi della finestra VSC.

L’Activity Bar:



* La prima icona mostra o nasconde la side bar
* La seconda gestisce la ricerca testuale nei file del folder e subfolder
* La terza gestisce GitHub e simili repository
* La quarta gestisce il debug
* La quinta gestisce le estensioni. Io ora ho le seguenti estensioni installate
  + Jupyter Notebook
  + Jupyter Slide Show
  + Notepad++ keymap
  + Pylance
  + Python
  + WSL
  + Docker

Le estensioni Python e Pylance sono fondamentali.

* La sesta gestisce le connessioni remote (per esempio per WSL, Windows Subsystem for Linux)
* La settima gestisce i Python test (se sono configurati)
* L’ottava se si vuole lavorare con Jupyter
* La nona gestisce Docker

# **Uso**

Il testo dei file Python appare in vari colori a seconda del comando o della variabile. Lo schema dei colori può essere cambiato (ma non è banale né consigliabile).

Non appena vengono commessi errori (cattiva indentazione, comandi mal scritti, variabili chiamate prima della loro definizione, moduli non importati, parentesi non chiuse, ecc.), vengono segnalati e a volte viene proposta una possibile correzione. A volte l’errore è solo “temporaneo”.

Quando cercate di scrivere il nome di variabile, vi suggerisce, e quando mettete il nome di una funzione e aprite la parentesi, vi spiega quali sono gli argomenti.

Possono tenersi aperte più finestre VSC, con settaggi diversi e su folder differenti.