Cheat Sheet

Le basi

print("Hello World")

#No dichiarazione

x= 3

#Tipizzazione variabile automatica

pi = 3.14

#isinstance restituisce un valore booleano se l'oggetto e il tipo sono uguali

isinstance(pi, int)

#input da tastiera tramite console con messaggio

z = input("Messaggio per l'input: ")

Contorllo di flusso

#indentazione fondamentale

if x == y:

    print("ramo true")

else:

    print("ramo false")

print("sono fuori dall'if")

if z<0:

    pass #non possiamo lasciare un blocco senza istruzioni, pass è un istruzione che non fa nulla

elif z == 0:

    print("uguale")

else:

    print("maggiore")

#ciclo while

contatore = 0

while contatore <=10:

    print(contatore)

    contatore +=1

#il do-while non è supportato nativamente, utilizzare un construtto di questo tipo

while True:

    y = int(input("inserisci 0 "))

    if y == 0:

        break

#Ciclo for

#importante la funzione range

#è composta da range(punto di inizio, punto di fine, quantità del passo)

for numero in range(2, 11, 2):

    print(numero)

Moduli esterni

import math

import random

#numeri random

random.random()

#funzioni matematiche base

print(math.sqrt(4))

Funzioni

def diIlMioNome():

    nome = input("Come ti chiami? ")

    print("Il tuo nome è: ", nome)

diIlMioNome()

#funzione con tipo di ritorno

#tipi dedotti in automatico dal compilatore

def addizione(a, b):

    risultato = a + b

    return risultato

Liste

#defininizione di una lista

lista = [2, "stringa", 1.11]

#enumerate

frutta = ["mela", "banana", "kiwi", "arancia"]

for indice, frutto in enumerate(frutta):

    print(indice, frutto)

Dizionari

#dizionari

#lista composta da coppie chiave valori

dizionario = {"chiave1" : "valore", "chiave2": 2}

#contorollare se c'è la chiave

print("chiave1" in dizionario)

#restiriusico tutte le chiavi

print(dizionario.keys())

#restiriusico tutti i valori

print(dizionario.values())

Classi e oggetti

class Persona:

    #costruttore

    def \_\_init\_\_(self, nome, cognome, età) -> None:

        #attributi

        self.nome = nome

        self.cognome = cognome

        self.età = età

    #metodi

    def NomeCognome(self):

print(self.nome + self.cognome)

    @staticmethod

    def somma(a,b):

        print(a+b)

#creazione istanza

valentino = Persona("valentino",

"rossi", 40)

valentino.NomeCognome ()

File di testo

#leggere

with open("File.txt","r") as file:

    for riga in file:

        print(riga)

#Scrivere

dati = ["dato1", "dato2", "dato3"]

with open("File.txt","w") as file:

    file.write("Inizio file")

with open("File.txt","a") as file:

    for dato in dati:

        file.write(dato)

        file.write("\n")

        print(dato)

Database

import textwrap

import pyodbc as db

#connessione ad un database

conn = db.connect('connectionString')

cursor = conn.cursor()

#Eseguire una select

n = 1

cursor.execute("select \* from Tabella1 where Campo1 = ?", n)

#prende solo il primo risultato

rows = cursor.fetchone()

#prende tutti i risultati

rows = cursor.fetchall()

#eseguire una insert, subito dopo usando la rollback

cursor.execute("INSERT INTO Tabella1 (Campo1, Campo2) VALUES (1, 'prova')")

#annulla le modifiche non committate

conn.rollback()

#chiudi la connessione

conn.close()