

1. Tipi di Dati

Numeri

- **int()**, **float()**, **complex()**: Conversione in numeri interi, decimali o complessi.
- **abs()**, **round()**, **pow()**: Valore assoluto, arrotondamento, potenza.
- **divmod()**, **sum()**: Divisione con resto, somma di una sequenza.

Stringhe

- **.lower()**, **.upper()**, **.title()**: Modifica il case della stringa.
- **.capitalize()**, **.strip()**, **.split()**: Capitalizza, rimuove spazi, divide in parole.
- **.join()**, **.replace()**, **.find()**: Unisce, sostituisce, cerca sotto-stringhe.
- **.format()**, **f-string**, **.count()**: Formatta, conta occorrenze.

Liste (List)

- **.append()**, **.extend()**, **.insert()**: Aggiunge elementi.
- **.remove()**, **.pop()**, **.clear()**: Rimuove o svuota la lista.
- **.index()**, **.count()**, **.sort()**, **.reverse()**: Trova, conta, ordina, inverte.

Tuples

- **.count()**, **.index()**: Conta e trova l'indice.

Dizionari (Dict)

- **.get()**, **.setdefault()**: Accesso sicuro, default per chiavi.
- **.keys()**, **.values()**, **.items()**: Restituisce chiavi, valori, coppie.
- **.update()**, **.pop()**, **.popitem()**: Aggiunge, rimuove, modifica.

Set

- **.add()**, **.remove()**, **.discard()**: Aggiunge e rimuove elementi.
- **.pop()**, **.clear()**: Rimuove un elemento e svuota.
- **.union()**, **.intersection()**, **.difference()**: Operazioni tra insiemi.

2. Funzioni Built-in

- **len()**, **type()**, **id()**: Lunghezza, tipo, ID oggetto.
- **sorted()**, **reversed()**: Ordina e inverte sequenze.
- **map()**, **filter()**, **reduce()**: Funzioni di trasformazione e aggregazione.
- **zip()**, **enumerate()**: Combina sequenze, numera gli elementi.
- **min()**, **max()**, **sum()**: Minimo, massimo, somma.
- **all()**, **any()**, **range()**: Verifica condizioni, range numerici.

- **input()**, **print()**: Input da utente, output su schermo.
-

3. Istruzioni Condizionali e Cicli

Condizionali

- **if, elif, else**: Controlla condizioni.
- **Operatori logici**: and, or, not per combinare condizioni.

Cicli

- **for, while**: Iterazione su sequenze o condizioni.
 - **break, continue**: Interrompe o salta un ciclo.
 - **else**: Codice eseguito se il ciclo non è interrotto.
-

4. Funzioni e Lambda

- **def**: Definisce una funzione.
 - **return**: Restituisce valore da una funzione.
 - **lambda**: Funzioni anonime in una riga.
 - ***args, **kwargs**: Argomenti variabili.
-

5. Gestione delle Eccezioni

- **try, except, else, finally**: Gestione di errori e pulizia finale.
 - **raise**: Solleva un'eccezione.
 - **assert**: Verifica condizioni, usato nei test.
 - **with**: Gestione sicura delle risorse.
-

6. Moduli e Pacchetti

- **import**: Importa moduli.
 - **from ... import ...**: Importa specifiche funzioni o variabili.
 - **as**: Alias per moduli o funzioni.
 - **__name__**: Verifica se il modulo è eseguito direttamente.
-

7. Decoratori e Generatori

Decoratori

- **@decorator**: Modifica il comportamento di una funzione.

Generatori

- **yield**: Crea generatori per iterazioni lente.
 - **next()**: Ottiene il prossimo elemento da un generatore.
 - **iter()**: Ottiene un iteratore da un oggetto.
-

8. Comprehension

- **List, Set, Dict Comprehension**: Creazione compatta di liste, set e dizionari.
 - **Generator Expressions**: Iterazioni lente per generare valori.
-

9. Funzioni di Classe e Oggetti

- **class**: Definisce una classe.
 - **__init__**: Costruttore per inizializzare oggetti.
 - **self**: Riferimento all'oggetto corrente.
 - **@classmethod, @staticmethod**: Metodi di classe e statici.
 - **__str__, __repr__**: Rappresentazioni stringa degli oggetti.
-

10. Altri Metodi Utili

- **del**: Elimina variabili o oggetti.
 - **globals(), locals()**: Ottiene i dizionari di variabili globali e locali.
 - **hash(), sorted(), reversed()**: Operazioni comuni.
 - **bytes(), bytearray()**: Gestione di dati binari.
 - **dir()**: Ottiene gli attributi di un oggetto.
 - **help()**: Documentazione interattiva.
-