

GABRIELE ROMANATO

Introduzione al type hinting in Python

Il type hinting è una funzionalità introdotta in Python 3.5 che consente agli sviluppatori di specificare i tipi di variabili, parametri e valori di ritorno delle funzioni. Sebbene Python rimanga un linguaggio dinamico, il type hinting migliora la leggibilità del codice e facilita strumenti di analisi statica come mypy, Pyright o Pyre.

Perché usare il Type Hinting?

- **Documentazione chiara:** i tipi rendono il comportamento del codice più esplicito.
- **Controlli statici:** gli strumenti possono rilevare errori prima dell'esecuzione.
- **Supporto IDE:** autocompletamento e suggerimenti diventano più precisi.

Come si utilizza il Type Hinting?

Si possono annotare i tipi di parametri e valori di ritorno delle funzioni. Ecco un esempio base:

```
def somma(a: int, b: int) -> int:  
    return a + b
```

In questo esempio, sia a che b sono definiti come interi, e la funzione restituisce un intero.

Tipi Complessi

Python fornisce diversi strumenti per descrivere tipi più complessi. È possibile utilizzare tipi come List, Dict, Tuple e Optional dal modulo typing:

```
from typing import List, Dict, Optional  
  
def processa_numeri(numeri: List[int]) -> Dict[str, int]:  
    return {"totale": sum(numeri)}  
  
def trova_valore(dati: Dict[str, int], chiave: str) -> Optional[int]:  
    return dati.get(chiave)
```

Type Alias e NewType

Per semplificare definizioni complesse, si possono usare alias o creare nuovi tipi:

```
from typing import List, NewType

UserId = NewType('UserId', int)

def get_user(user_id: UserId) -> str:
    return f"User {user_id}"
```

Type Hinting in Python 3.9+

Con Python 3.9 e versioni successive, è possibile utilizzare una sintassi più semplice, eliminando la necessità di importare alcuni tipi da typing:

```
def unisci_stringhe(elementi: list[str]) -> str:
    return ''.join(elementi)
```

Conclusioni

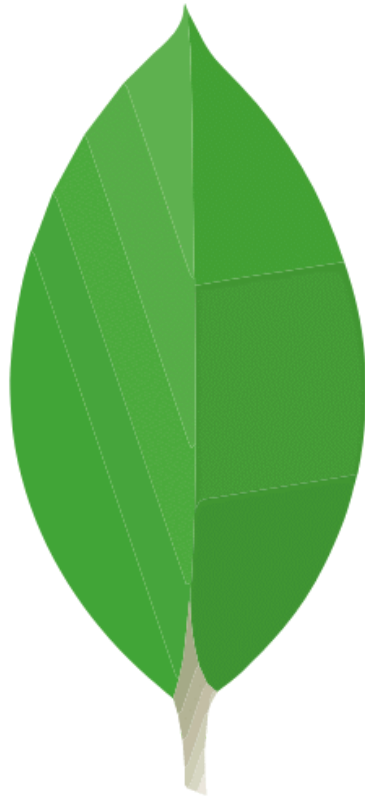
Il type hinting rappresenta un potente strumento per migliorare la qualità del codice Python. Pur non essendo obbligatorio, il suo utilizzo è fortemente consigliato nei progetti di medie e grandi dimensioni.

Applicazioni Correlate



Python Placeholder Image

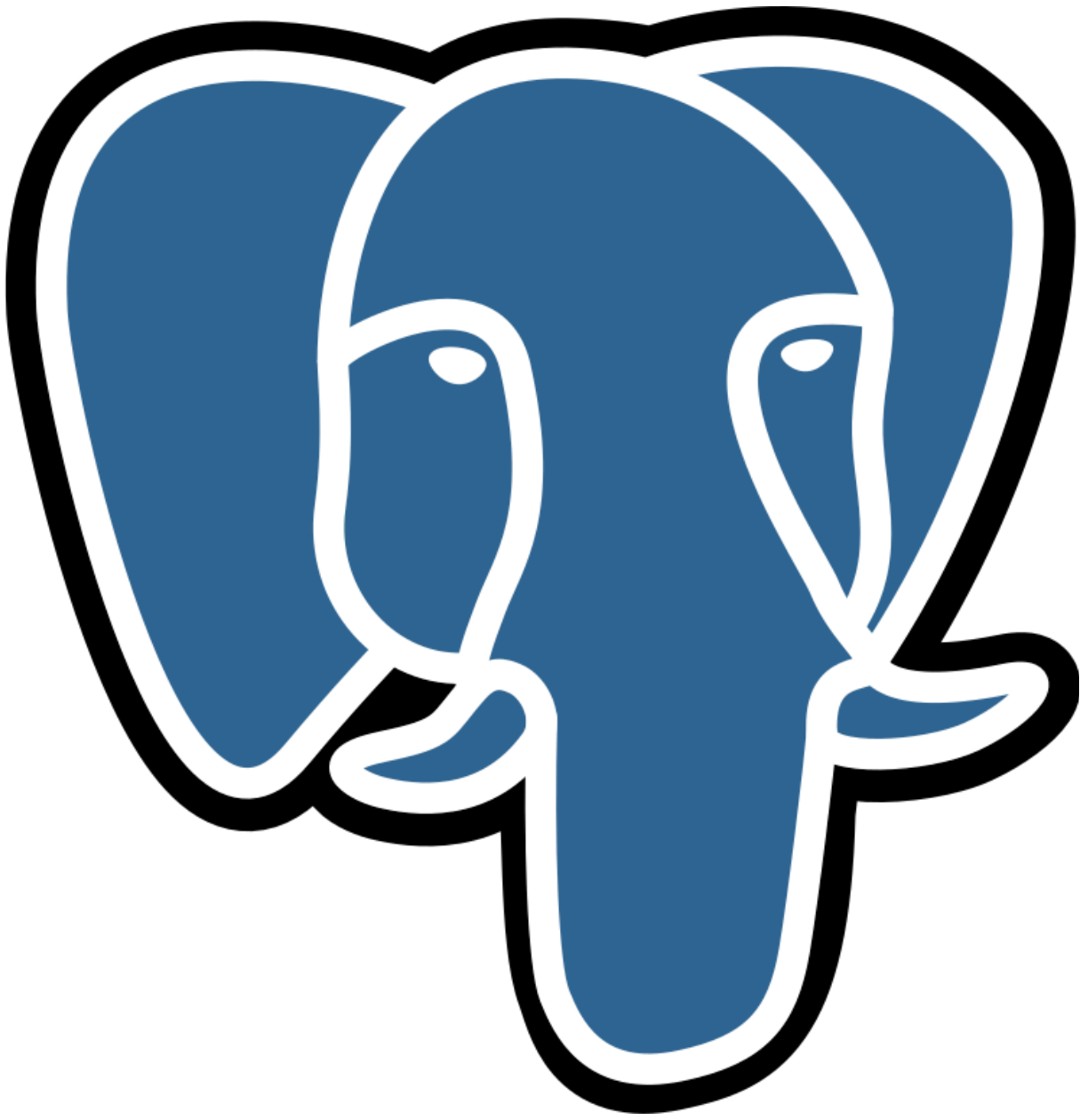
Applicazione sviluppata in Python con Flask per la creazione di immagini segnaposto.
Docker Docker Compose Python Flask JavaScript



-

Python MongoDB App

Applicazione basata su MongoDB e sviluppata in Python e Flask.
DockerDocker ComposePythonFlaskMongoDB



•

Python PostgreSQL App

Applicazione basata su PostgreSQL con Python e Flask.

Docker Docker Compose Python Flask