

Modern Python in FastAPI - Prima parte



Type Hints: usati non solo documentare ma anche per conversioni automatiche

Classi Pydantic -> Data classes + validazioni

... alla base di FastAPI e SQLAlchemy



Type Hints

Type hinting è una soluzione formale per indicare staticamente i tipi nel codice Python. Specificati in PEP 484 e introdotti dalla versione Python 3.5.

Utile cheat sheet per Type Hints

https://mypy.readthedocs.io/en/latest/cheat_sheet_py3.html

Type hints applied (esempio)

Variabili e funzioni con/senza type hints

```
costo_massimo = None
vs
costo_massimo: Optional[int] = None

def calcola_costo(elementi):
vs
def calcola_costo(elementi: Iterable[Elemento]) -> int:
```

Comunicazione e leggibilità

Diversa gestione di: *e.prezzo* nelle hints dell'editor



Pydantic

<https://pydantic-docs.helpmanual.io/>

```
from pydantic import BaseModel
```

Validazione dati e gestione valori usando le Python type annotations

pydantic applica type hints in fase di esecuzione e fornisce errori di facile utilizzo quando i dati non sono validi.

Definisci come dovrebbero essere i dati in puro Python; convalidalo con pydantic.

Per approfondimenti su Pydantic vs dataclasses:

<https://www.youtube.com/watch?v=vRVVyl9uaZc> (dataclasses)

https://www.youtube.com/watch?v=Vj-iU-8_xLs&t=3s (pydantic vs dataclasses)

Pydantic in validazione

Sanificazione dei dati

Esempio