

Welcome to Pydantic

 CI passing

 coverage 96%

 pypi v2.13.4

 conda | conda-forge v2.13.4

 downloads/month 1G

 license MIT

 llms.txt

Documentation for version: v2.13.4.

Pydantic is the most widely used data validation library for Python.

Fast and extensible, Pydantic plays nicely with your linters/IDE/brain. Define how data should be in pure, canonical Python 3.9+; validate it with Pydantic.

Sign up for our newsletter, *The Pydantic Stack*, with updates & tutorials on Pydantic, Logfire, and Pydantic AI:

[Subscribe](#)

Why use Pydantic?

- **Powered by type hints** — with Pydantic, schema validation and serialization are controlled by type annotations; less to learn, less code to write, and integration with your IDE and static analysis tools. [Learn more...](#)
- **Speed** — Pydantic's core validation logic is written in Rust. As a result, Pydantic is among the fastest data validation libraries for Python. [Learn more...](#)
- **JSON Schema** — Pydantic models can emit JSON Schema, allowing for easy integration with other tools. [Learn more...](#)
- **Strict and Lax mode** — Pydantic can run in either strict mode (where data is not converted) or lax mode where Pydantic tries to coerce data to the correct type where appropriate. [Learn more...](#)
- **Dataclasses, TypedDicts and more** — Pydantic supports validation of many standard library types including dataclass and TypedDict. [Learn more...](#)

- **Customisation** — Pydantic allows custom validators and serializers to alter how data is processed in many powerful ways. [Learn more...](#)
- **Ecosystem** — around 8,000 packages on PyPI use Pydantic, including massively popular libraries like *FastAPI*, *huggingface*, *Django Ninja*, *SQLModel*, & *LangChain*. [Learn more...](#)
- **Battle tested** — Pydantic is downloaded over 550M times/month and is used by all FAANG companies and 20 of the 25 largest companies on NASDAQ. If you're trying to do something with Pydantic, someone else has probably already done it. [Learn more...](#)

[Installing Pydantic](#) is as simple as: `pip install pydantic`

Pydantic examples [🔗](#)

To see Pydantic at work, let's start with a simple example, creating a custom class that inherits from `BaseModel`:

Validation Successful

```
from datetime import datetime

from pydantic import BaseModel, PositiveInt

class User(BaseModel):
    id: int
    name: str = 'John Doe'
    signup_ts: datetime | None
    tastes: dict[str, PositiveInt]

external_data = {
    'id': 123,
    'signup_ts': '2019-06-01 12:22',
    'tastes': {
        'wine': 9,
        'cheese': 7,
        'cabbage': 1,
    },
}

user = User(**external_data)

print(user.id)
#> 123
print(user.model_dump())
"""
{
    'id': 123,
    'name': 'John Doe',
    'signup_ts': datetime.datetime(2019, 6, 1, 12, 22),
    'tastes': {'wine': 9, 'cheese': 7, 'cabbage': 1},
}
"""
```

If validation fails, Pydantic will raise an error with a breakdown of what was wrong:

Validation Error

continuing the above example...



```
from datetime import datetime
from pydantic import BaseModel, PositiveInt, ValidationError
```

```
class User(BaseModel):
    id: int
    name: str = 'John Doe'
    signup_ts: datetime | None
    tastes: dict[str, PositiveInt]
```

```
external_data = {'id': 'not an int', 'tastes': {}} +
```

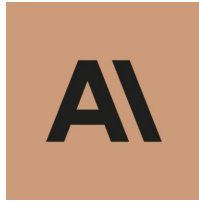
```
try:
    User(**external_data) +
except ValidationError as e:
    print(e.errors())
    """
    [
        {
            'type': 'int_parsing',
            'loc': ('id',),
            'msg': 'Input should be a valid integer, unable to parse string as an integer',
            'input': 'not an int',
            'url': 'https://errors.pydantic.dev/2/v/int_parsing',
        },
        {
            'type': 'missing',
            'loc': ('signup_ts',),
            'msg': 'Field required',
            'input': {'id': 'not an int', 'tastes': {}},
            'url': 'https://errors.pydantic.dev/2/v/missing',
        },
    ]
    """
```

Who is using Pydantic? [🔗](#)

Hundreds of organisations and packages are using Pydantic. Some of the prominent companies and organizations around the world who are using Pydantic include:



amazon



ASML

AstraZeneca



Capital One



IBM



INTUIT



JPMORGAN
CHASE & CO.



ORACLE



Qualcomm



Revolut



For a more comprehensive list of open-source projects using Pydantic see the [list of dependents on github](#), or you can find some awesome projects using Pydantic in [awesome-pydantic](#).

Was this page helpful?

