



Hamilton Global User Group October 2024 Meetup

What is Hamilton?

Hamilton helps data scientists and engineers define testable, modular, self-documenting dataflows, that encode lineage and metadata.

Runs and scales everywhere python does.

Icebreaker: Name and what you're using Hamilton for/looking for.

Agenda

1. Community Spotlight
2. The "news"
3. Open 



Community Spotlight:
Sholto Armstrong



The "News"

Some Recent Features / Improvements Released

- Pydantic Validator (thanks Charles Schwartz)
- Static Graph Validation
- @mutate & @pipe_output (thanks Jernej Frank)
- Caching (thanks Thierry Jean)

Pydantic Validator

```
from hamilton.function_modifiers import check_output
from hamilton.plugins import h_pydantic
```

```
class MyModel(BaseModel):
    name: str
```

```
@check_output(model=MyModel)
def foo() -> dict:
    return {"name": "hamilton"}
```

```
# or
```

```
@h_pydantic.check_output()
def foo() -> MyModel:
    return MyModel(name="hamilton")
```

Static Graph Validation

```
from typing import Optional
from hamilton.lifecycle import api
from hamilton.graph_types import HamiltonNode

class MyTagValidator(api.StaticValidator):
    """Static validator that is run right after the graph is built."""

    def run_to_validate_node(
        self, *, node: HamiltonNode, **future_kwargs
    ) -> tuple[bool, Optional[str]]:
        """Validates that if a node has one tag, it has another one as well"""
        if node.tags.get("node_type", "") == "output":
            table_name = node.table_name
            if not table_name:
                from hamilton import driver
                error_msg = (f"
                    " dr = (
                        driver.Builder()
                            .with_modules(my_module)
                            .with_adapters(MyTagValidator()) # <-- add here
                            .build()
                    )
                return False, error_msg
            return True, None
```

@mutate & @pipe_output

Caching



Next meet-up - December:

Speaker? Want to speak?



Open Mic.

FIN. Thanks for coming!

