Typed HTML in *my* Python?

2025-02-01 Brussels

Athena Wolfskämpf

Tech Lead Germany @ Personalkollen

Scheduled Stations

- Motivations for typed templating
- `htpy` in action
- Patterns for success

This talk won't call today at

- General overview of typing in Python
- Step by step guides (look at the `htpy` documentation instead)

Motivations

- Static type checks!

```
aw@ath3na ~/personalkollen (main)> mypy
Success: no issues found in 1625 source files
```

- Avoid errors discovered only at runtime
- Make use of existing refactoring tools

- Does not introduce a new template language
- It's Python!
- Does not care about typing unless you want it to
- Used in production at Personalkollen, serving 100k users in Sweden
- https://htpy.dev/ [QR Code ➡️]

Syntax

- HTML attributes are specified by parenthesis (`()` / "call")
- Children are specified using square brackets (`[]` / "getitem")

```
>>> import htpy as h
>>> print(h.p(id="greeting")["Hello!"])
<p id="greeting">Hello!</p>
```

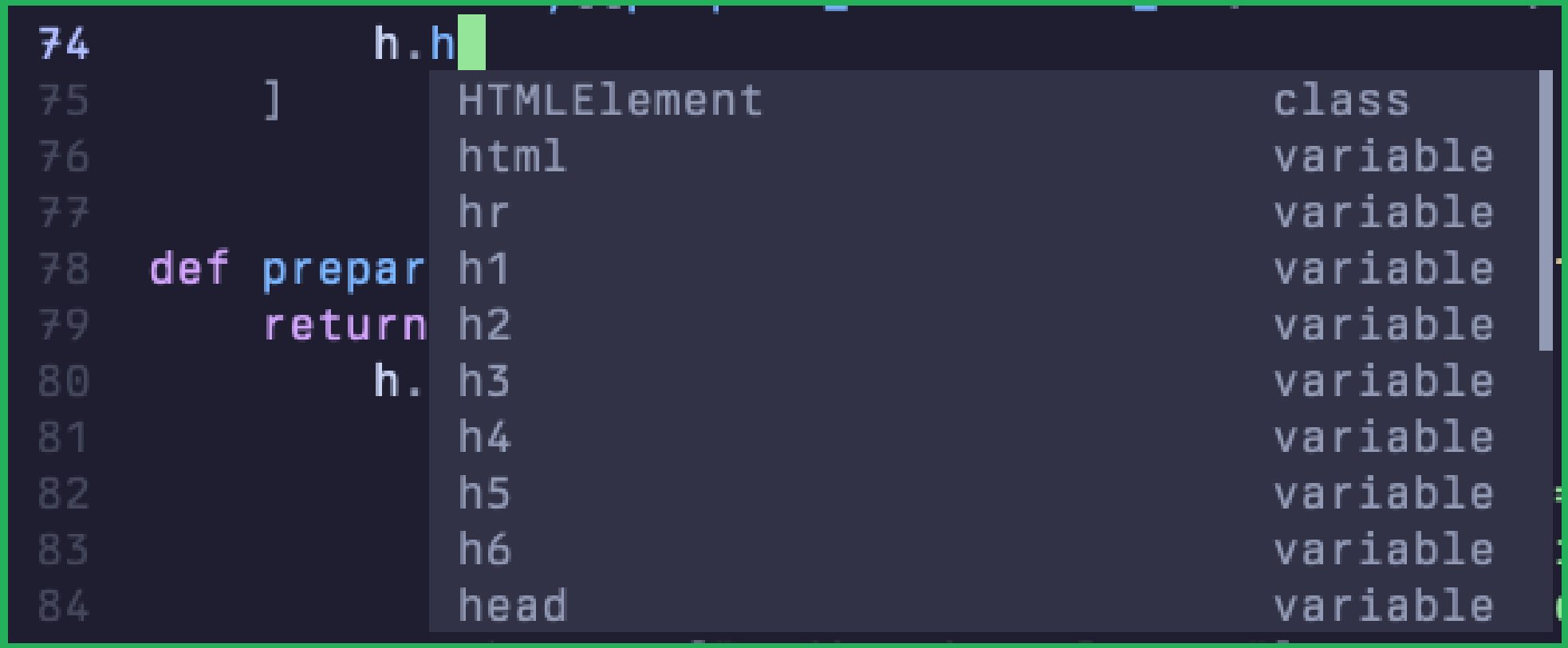
Syntax

- HTML attributes are specified by parenthesis (`()` / "call")
- Children are specified using square brackets (`[]` / "getitem")

```
>>> import htpy as h
>>> print(h.p("#greeting.large")["Hello!"])
<p id="greeting" class="large">Hello!</p>
```

Autocomplete for all the HTML tags you love

- `htpy` comes pre-loaded with all standardised HTML tags for autocomplete



Import anything, even custom HTML tags

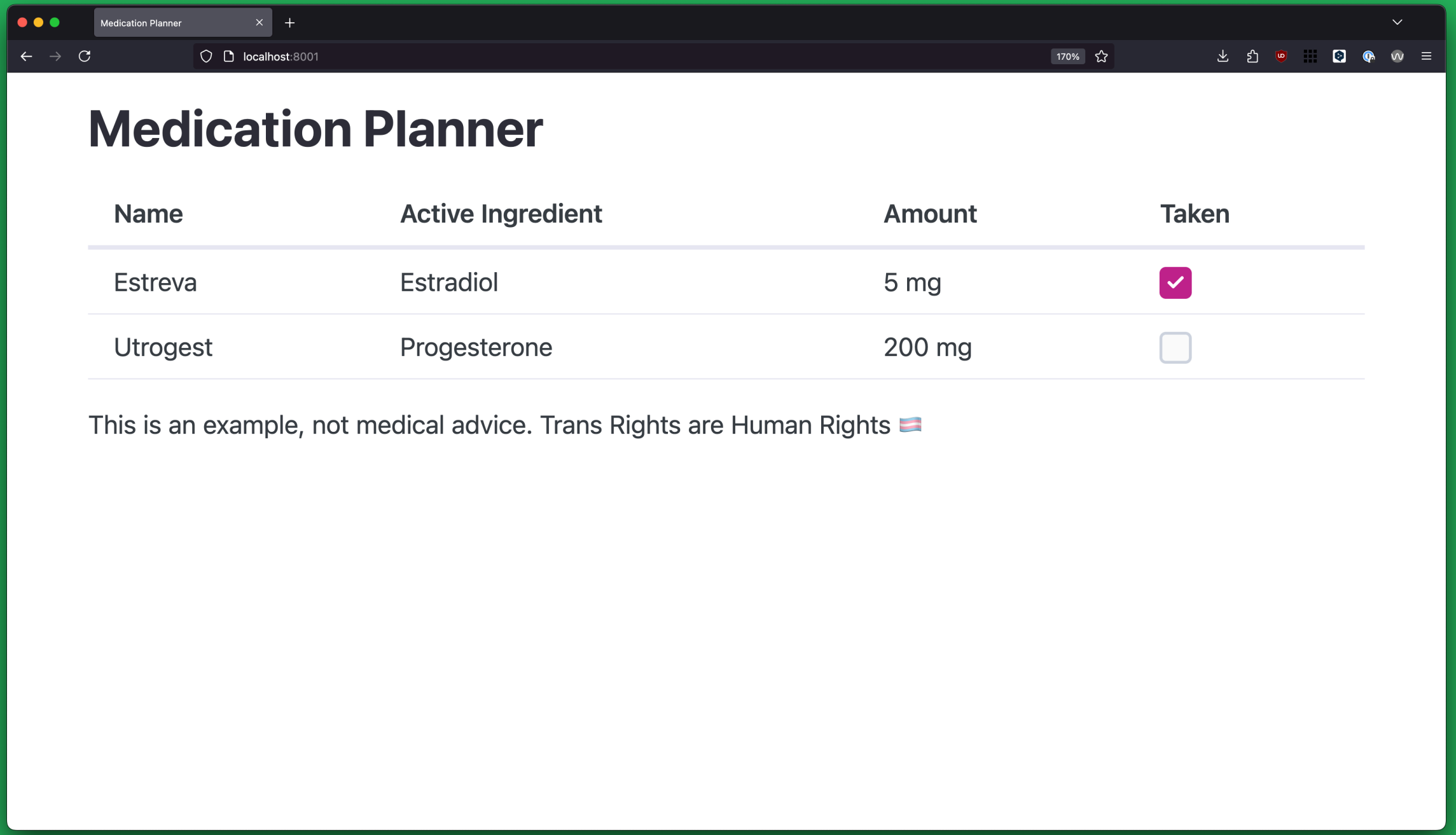
- Web Components? No Problem!

```python
# htpy.py

def __getattr__(name: str) -> Element:
    # Simplified implementation without caching and validation
    return Element(name.replace("_", "-"))
```

```
>>> import htpy as h
>>> print(h.sl_alert(open=True)["Please talk to your pharmacist."])
<sl-alert open>Please talk to your pharmacist.</sl-alert>
```

Patterns for success and Demo


Medication Planner
localhost:8001
170%
Medication Planner
Name
Active Ingredient
Amount
Taken
Estreva
Estradiol
5 mg
Utrogest
Progesterone
200 mg
This is an example, not medical advice. Trans Rights are Human Rights 🏳️‍⚧️

```python
class Unit(enum.StrEnum):
    mg = "mg"
    g = "g"


@dataclasses.dataclass()
class ActiveIngredient:
    name: str
    amount: decimal.Decimal
    unit: Unit


@dataclasses.dataclass()
class Medication:
    name: str
    active_ingredient: ActiveIngredient
    manufacturer: int  # pretend this is a ForeignKey
```

Use native language features

```
def medication_plan(medications: list[Medication]) -> h.Element:
    return h.table[
        h.thead[
            h.tr[
                h.th(scope="column")["Name"],
                h.th(scope="column")["Active Ingredient"],
                h.th(scope="column")["Amount"],
                h.th(scope="column")["Taken"],
            ]
        ],
        # Generator expression! 🤯
        h.tbody[(medication_tr(medication) for medication in medications)],
    ]
```

Use the full power of static typing and your LSP

- Autocomplete for attributes
- Static type checking ensures correctness

```
def medication_tr(medication: Medication) -> h.Element:
    return h.tr[
        h.td[medication.name],
        h.td[medication.active_ingredient.name],
        h.td[
            f"{medication.active_ingredient.amount} {medication.active_ingredient.unit}"
        ],
        h.td[h.input(type="checkbox")],
    ]
```

Base template is also a method

- Generated this with `html2htpy` from the Pico CSS base template

```
def base(main_content: h.Node) -> h.Element:
    return h.html(lang="en")[
        h.head[
            h.meta(charset="utf-8"),
            h.meta(name="viewport", content="width=device-width, initial-scale=1"),
            h.meta(name="color-scheme", content="light dark"),
            h.link(rel="stylesheet", href="static/css/pico.fuchsia.min.css"),
            h.title["Medication Planner"],
        ],
        h.body[h.main(".container")[main_content]],
    ]
```

Usage of these components in Starlette

- The same, unchanged components can be used in Django, FastAPI, Flask and as shown: Starlette

```
async def index(request: Request) -> HTMLResponse:
    return HTMLResponse(
        base(
            main_content=[
                h.h1["Medication Planner"],
                medication_plan(medication_table),
                h.p[
                    "This is an example, not medical advice. Trans Rights are Human Rights."
                ],
            ]
        )
    )
```

- https://htpy.dev/ [QR Code ➡️]
- Get in touch via GitHub discussions

Athena Wolfskämpf

- Socials: https://wolfskaempf.de [QR Code ➡️]
- Talk to me at FOSDEM
- Fediverse: @wolfskaempf@climatejustice.social

References

- Photos: Unsplash licensed under the Unsplash License
- Project logos belong to the respective project
- Code examples licensed under MIT