Best Practices for Writing Code for Other People to Use

Josh Schneider · github/dijital20 · @diji@twitter.com · @diji@mastodon.social

PyTexas Meetup, November 2023

What?!?

- **Who are "others"?**
    - Obviously, people **that are not you**.
    - **You 6 months** from now is different than **you now**.
- **What do you mean by "use"?**
    - Installs and includes
    - Maintains and contributes

> "*Always code as if the person maintaining your code will be a violent psychopath who knows where you live.*"
>
> *-- Martin Golding*

Josh's 3 (make that 4) rules for good UX design

- Rule 1: Make it **obvious**
- Rule 2: Make it **familiar**
- Rule 3: Make it **well-documented**
- Rule 4: Start at Rule 1, not Rule 3.

About Style

> *"I don't care what you believe in, just believe in it."*
>
> *-- Shepherd Book, "Serenity"*

- Document it (Consider a `STYLE.md` file)
- Apply it consistently

Design

- Use SOLID principles for classes/objects
- Avoid classes when you don't need them
- Use good naming
  - Names should be descriptive to purpose.
  - Verb names for functions/methods - *Functions do!*
  - Noun names for variables/fields - *Variables are!*

In-code Documentation

- Use type hints
- Write docstrings for all modules, classes, functions, and methods.
  - Summary
  - Input arguments and Output
  - Other

Example Docstring

```python
def write_results_to_file(results: list[dict[str, str | int]], path: str | Path) -> Path:
    """Writes the result structure to a file.

    Args:
        results: Data structure to write as a list of dictionaries.
        path: Path of the file to write to.

    Returns:
        Path of the file written.

    Raises:
        TypeError: If results is not a list of dicts with string keys.
    """
    ...
```

Line comments

Add context or justification to the code

```
# For some reason, the doohickey.do_thing function needs the last byte
# cut off. This is not documented in their libarary.
```

Signposts on where to find things

```
# See DESIGN.md, section "Gnarly Function"
```

Supplemental Documentation

- Text/Markdown documentation
- Unit tests
- Sample code

Thanks!