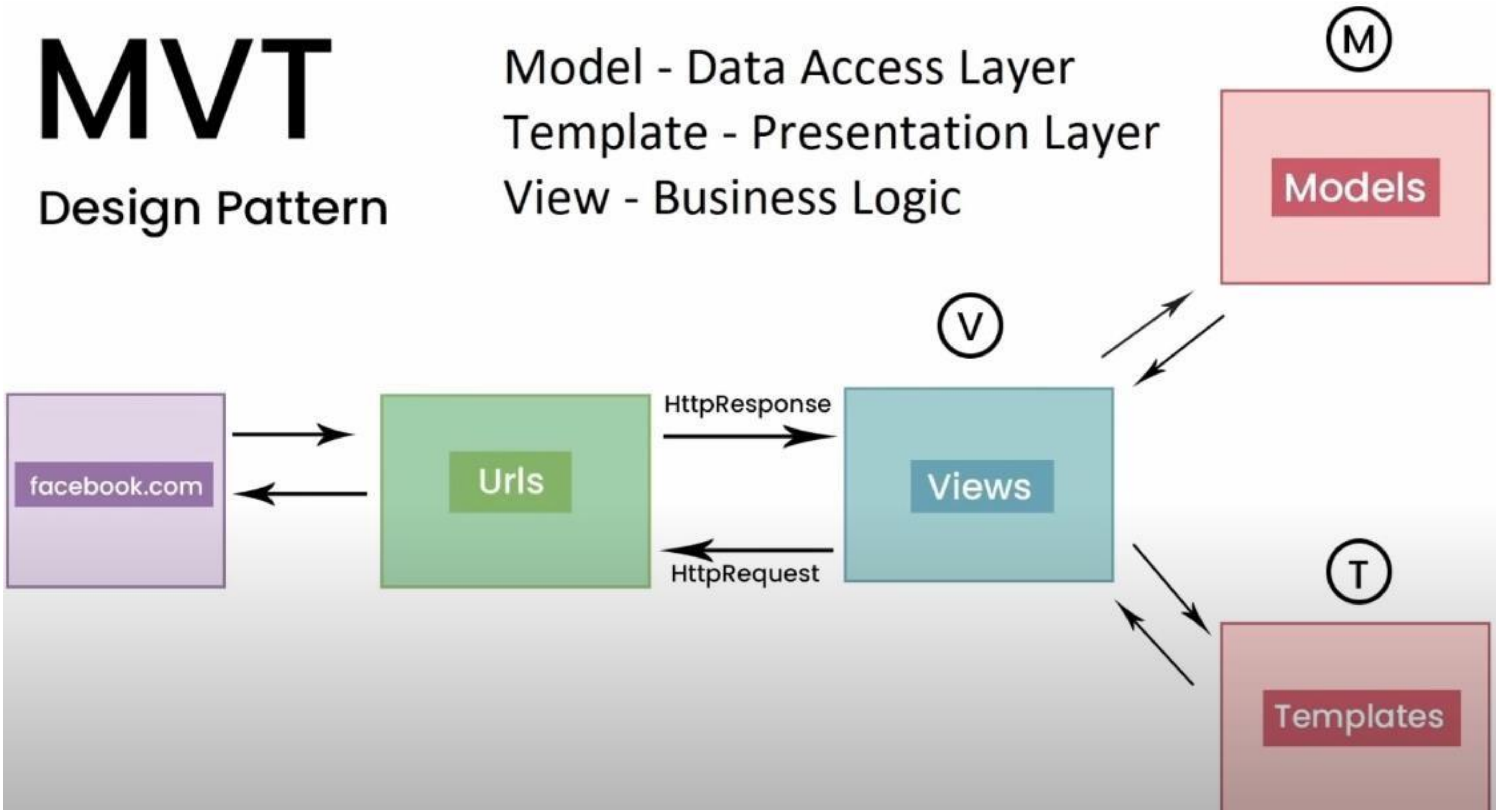


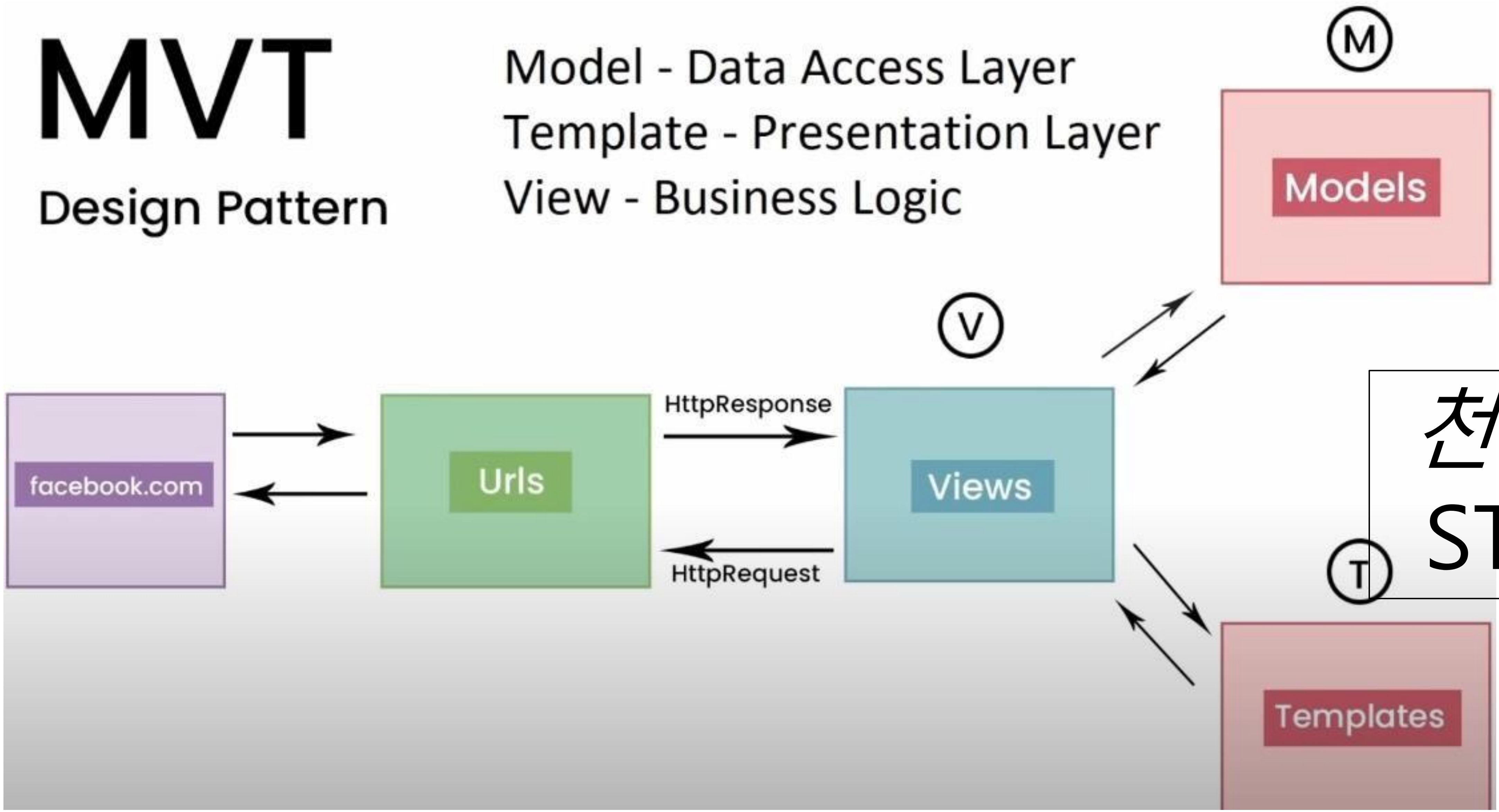
MVT

Design Pattern

Model - Data Access Layer

Template - Presentation Layer

View - Business Logic



MVT

Design Pattern

Model - Data Access Layer
Template - Presentation Layer
View - Business Logic

천재
STT

업무 분류

- 사용자에게 임의의 문장을 보여준 후
- 사용자의 음성을 받아서
- 천재 STT를 거쳐서
- 두 문장을 비교(Levenshtein)하여
- 그 결과를 DB에 저장

- "기존의 문장"과 "음성을 받아서 STT를 거친 문장"를 비교하는 코드를 짜는 role
- Django로 이를 포장하는 코드를 짜는 role

한계

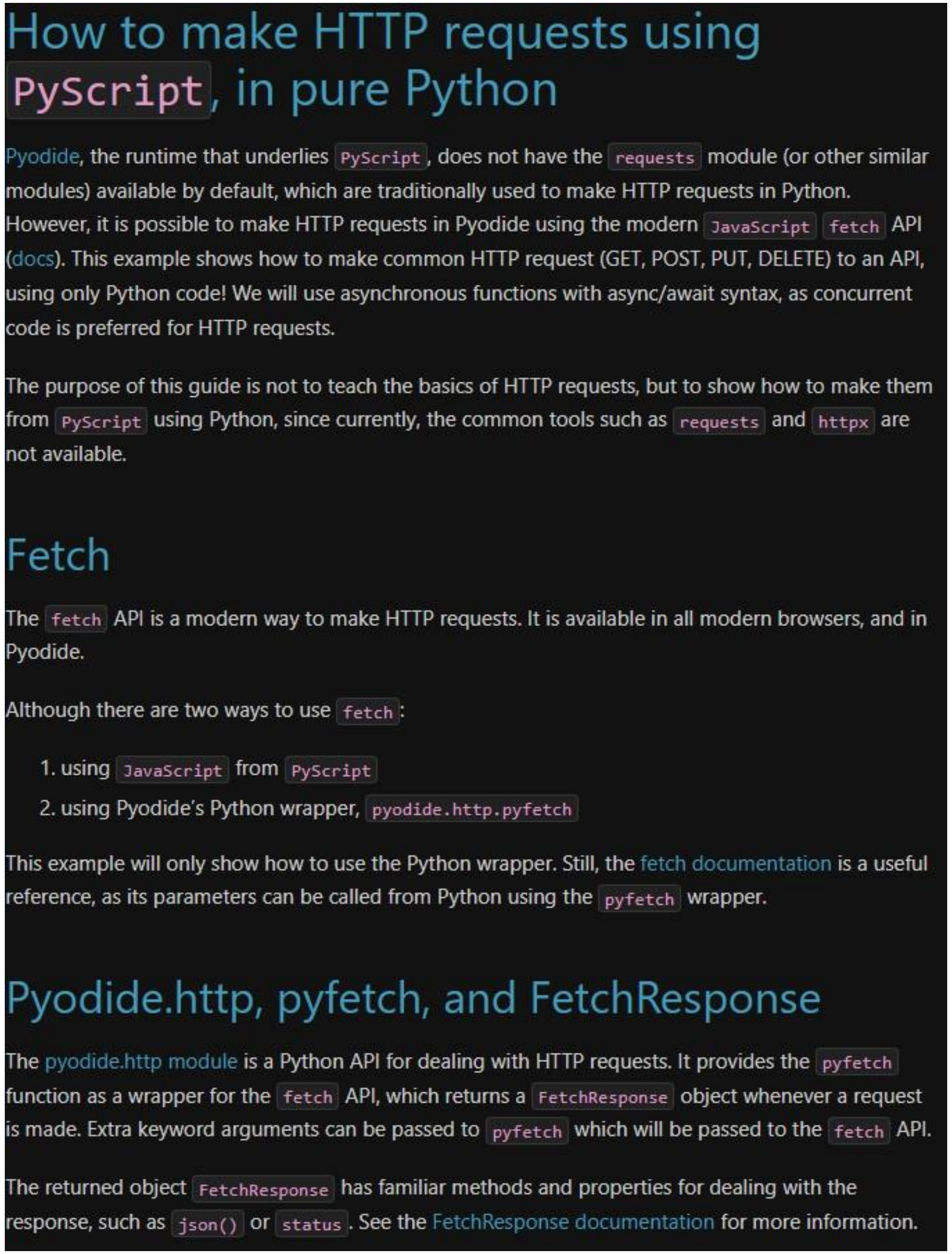
- 장고를 이용하여 해결하려고 했지만, Docker, AWS 아키텍처 구성, 이미지처리 수업을 들으면서, Framework를 학습한다는 것은 너무 힘들었다.
- Django에 대한 양질의 한국어 교육 컨텐츠가 부족했다.

- Django가 아닌 PyScript를 이용하여 프론트를 구성하려고 하였다.
- 하지만, HTTP를 이용한 파이썬 코드를 PyScript로 구현하기 위해서는 특수한 작업이 필요하다.

PyScript

```
<html>
  <py-script>
    print('이게 진짜 되네요')
  </py-script>
</html>
```

https://www.youtube.com/watch?v=3DuyJf_XPtM https://docs.pyscript.net/latest/guides/http-requests.html



How to make HTTP requests using `PyScript`, in pure Python

Pyodide, the runtime that underlies `PyScript`, does not have the `requests` module (or other similar modules) available by default, which are traditionally used to make HTTP requests in Python. However, it is possible to make HTTP requests in Pyodide using the modern `JavaScript` `fetch` API (docs). This example shows how to make common HTTP request (GET, POST, PUT, DELETE) to an API, using only Python code! We will use asynchronous functions with async/await syntax, as concurrent code is preferred for HTTP requests.

The purpose of this guide is not to teach the basics of HTTP requests, but to show how to make them from `PyScript` using Python, since currently, the common tools such as `requests` and `httpx` are not available.

Fetch

The `fetch` API is a modern way to make HTTP requests. It is available in all modern browsers, and in Pyodide.

Although there are two ways to use `fetch`:

1. using `JavaScript` from `PyScript`
2. using Pyodide's Python wrapper, `pyodide.http.pyfetch`

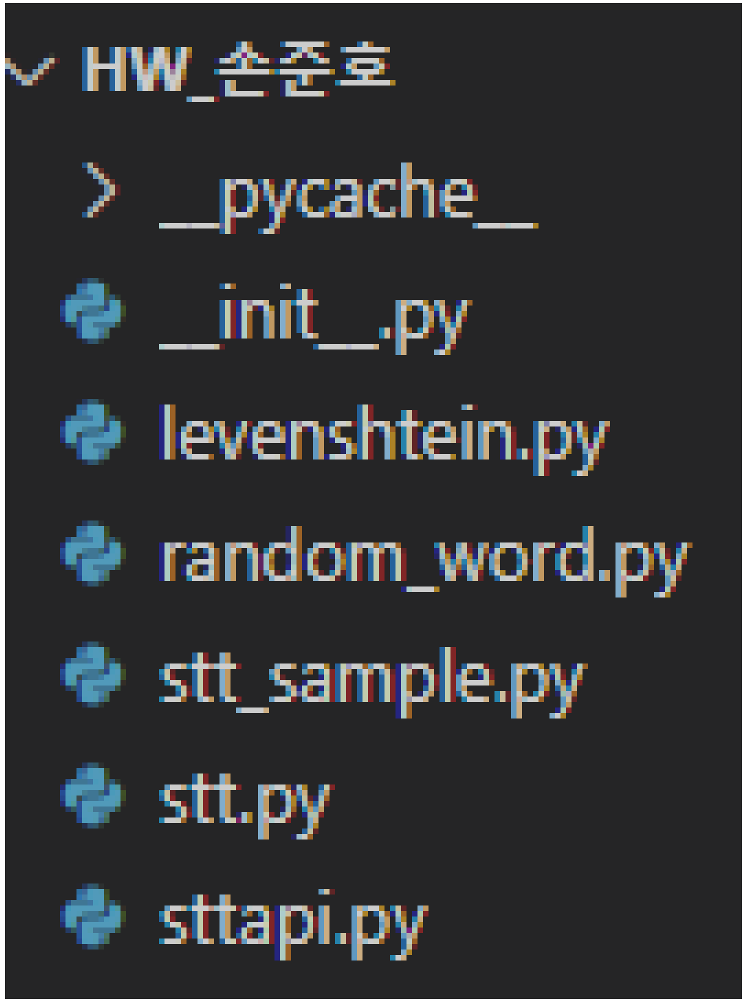
This example will only show how to use the Python wrapper. Still, the fetch documentation is a useful reference, as its parameters can be called from Python using the `pyfetch` wrapper.

Pyodide.http, pyfetch, and FetchResponse

The pyodide.http module is a Python API for dealing with HTTP requests. It provides the `pyfetch` function as a wrapper for the `fetch` API, which returns a `FetchResponse` object whenever a request is made. Extra keyword arguments can be passed to `pyfetch` which will be passed to the `fetch` API.

The returned object `FetchResponse` has familiar methods and properties for dealing with the response, such as `json()` or `status`. See the FetchResponse documentation for more information.

미완의 작품



HW_손준호
- __pycache__
- __init__.py
- levenshtein.py
- random_word.py
- stt_sample.py
- stt.py
- sttapi.py