

Python and Javascript for LLM Applications

Learn only what you need

Introduction

Python and LLM Apps

- Most AI frameworks are written in Python.
- We can use Python for server-side app development.
- FastAPI.

JS and LLM Apps

- Web applications.
- Mostly frontend, also backend.
- React, Next.js, Vercel.

Python syntax

- No ;
- Indentation.
- Variables not declared.

JS syntax

- Yes ;
- Declare variables, or not.

Python execution

- Jupyter notebooks.
- Code editor.

JS execution

- Jupyter notebooks.
- Browser's console.
- Code editor.

Python data types

- Arrays are called Lists.
- Objects are called Dictionaries.
- Tuples: unchangeable lists.

JS data types

- Arrays.
- Objects.

Python functions

- Declared with `def`.
- Peculiar: `*args` & `**kwargs`.
- Peculiar: lambda functions.
- Peculiar: lambda function inside regular function.

JS functions

- Declared with `function`.
- Peculiar: arrow functions.
- Peculiar: arrow function inside another arrow function.

Python Classes & Objects

- OO language: everything is an O.
- Declared with class
- `__init__`
- `self`
- `__str__`: string representation
- Empty class: use `pass`
- Object declared with `Class()`

JS Classes & Objects

- `.`
- Declared with class
- constructor
- `this`
- `.`
- `.`
- Object declared with `new Class()`

Python Methods & Props

- Nothing peculiar

JS Methods & Props

- Nothing peculiar

Python Inheritance

- `class Student(Person)`

JS Inheritance

- `class Student extends Person`

Python modules

- import

JS modules

- import

Python packages

- pip install
- pypi.org

JS packages

- npm install
- npmjs.com

Python debugging

- Try... except
- Raise exception

JS debugging

- Try... catch
- Throw new Error
- Browser's console

Python: other

- User input: `input()`

JS: other

- User input: `prompt()`

Python: know where

- Python JSON
- File handling
- Tutorials
 - Scikit
 - Numpy
 - Pandas
 - Matplotlib
 - Machine Learning

JS: know where

- JS Async
- JS JSON