**학습내용 정리**

1. hidden class

\* 컴퓨터 OS(operating system, 운영체제)

- 우리의 프로그램이 동작할 수 있는 구동 환경

- 프로그램은 OS에 의존적 🡪 OS에 맞춰서 개발 필요 🡪 어떤 개발 환경에서 개발을 실행할 것인가에 대한 선택

\* 파일 시스템 (File System)

- OS에서파일을저장하는트리구조저장체계

- 파일 vs 디렉토리

- 파일(File)

- 컴퓨터에서 정보를 저장하는 논리적 단위

- 파일명과 확장자로 식별

- 실행 O, 쓰기 O, 읽기 O

- 디렉토리(Directory) \*=\* 폴더 (\*=\* :: 비슷하다의 의미)

- 파일과 다른 디렉토리 포함 O

- 트리구조(root 디렉토리 🡪 루트 노드)

- 경로 vs 절대 경로 vs 상대 경로

- 경로 : 컴퓨터 파일 고유 위치 / 노드의 연결

- 절대 경로 : 루트 디렉토리 ~ 파일위치 까지의 경로

- 상대 경로 : 현재 디렉토리 ~ 타깃 파일까지 경로

\* 터미널(Terminal)

- 키보드로 명령을 입력 프로그램 실행

- CLI(Command Line Interface)

- 텍스트(Text)를 사용하여 컴퓨터에 명령 입력하는 인터페이스 체계

2. Python

\* 특징

- 플랫폼 독립적

- 플랫폼 = OS(ex. 윈도우, 리눅스, 안드로이드, 맥OS, iOS 등)

- 독립적인 = 관계없는, 상관없는 🡪 OS에 상관없이 한번 프로그램을 작성하면 사용가능

- 인터프리터 언어 = 통역기를 사용하는 언어

- 소스코드를 바로 실행할 수 있게 지원하는 프로그램 실행 방법

- 객체 지향

- 객체 지향적 언어

- 실행 순서가 아닌 단위 모듈(객체) 중심으로 프로그램을 작성

- 하나의 객체는 어떤 목적을 달성하기 위한 행동(method)와 속성(attribute)을 가지고 있음

- 동적 타이핑 언어

- 프로그램이 실행하는 시점에 프로그램이 사용해야 할 데이터에 대한 타입을 결정함

\* 장점

- 쉽다 / 간단 / 다양

- 이해하기 쉬운 문법

- 다양한 라이브러리 🡪 무엇을 생각하든 그것을 구현O

- 이미 널리 쓰임 🡪 어디에든 쓸 수 있는 언어

**피어세션 내용 정리**

4. 피어규칙

> 상식선에서 서로 무례하지않게 상호 존중

> 2시까지 늦지않게 입장

> 불가피한 사유로 입장하기 어려울 경우 모더레이터에게 개인메시지나 slack 조별 채널에 말하기

5. 피어플랜

- (핵심) 아침 강의 들은 후 심도 있게 토론

> 강의중 모르는 부분에 대해 질문하고 생각해보는 시간

> 다양한 전공자들이 모인 만큼 자신있는 부분에 대해서 알려줬으면 좋겠다는 내용, 부족한 부분에 대해서 알고싶은 내용 등이 제기되면 따로 하루 시간을 마련하여 준비자가 발표하고 토론

> 퀴즈 대비 핵심내용 이야기

> 유용한 자료, 유용한 지식 서로 공유

> 시간이 지나고 토론하면서 여러가지 좋은 방식들이 있으면 추가될수있음. 형식에 구애받지 않을 것

> 별도로 8주내에 간단한 AI 프로젝트 하나 정도 수행하거나 매주 지정된 알고리즘 문제 풀이 토론할 계획도 이야기가 나옴