

JARAM 하계 워크숍

Discord Bot 만들기

왜 이 주제를 선택하였는가?

Q: 왜 디스 코드를
이용하는 주제인가?

Q: 난이도는 어떻게
되는가?

Q: 무슨 언어를
사용하는가?

A: 현재 자람에서
다양한 용도로 디스
코드를 사용하기
때문에 선택했다.

A: 다른 Python
프로젝트(ex. 머신
러닝)보다 비교적 쉬운
편으로 선택을 했다.

A: 1학년은 Python을
배웠기 때문에
Python을 위주로
주제를 선택했다.

Discord란?

- 인스턴트 메신저
- 메시지, 보이스 채팅, 영상 공유 등
- 한국에서는 온라인 게임 즐기는 사람들이 많이 이용
- 개발자 사이에서도 사용



Making Discord Bot

- 팀 프로젝트
- Python (팀원 모두 동의 하에 다른 언어 가능)
- 오픈 소스 프로젝트 (discord.py) (Python 3.5 이상)
- 빈 프로젝트 or 오픈 소스 프로젝트 이어 가기

cf. Discord API ?

HOW?

개발을 진행을 할 것인가

- Goorm IDE (실시간 개발 공유)
- Visual Studio Code Live Share 이용
- Github

goormide

Live Coding
in VS Code



Making Discord Bot Functions

- ROLE
- VOICE (MUSIC, VIDEO STREAMING)
- MESSAGE (EMOJI, GAMEPLAY, JPG, REACTION, etc.)
- WEBHOOK

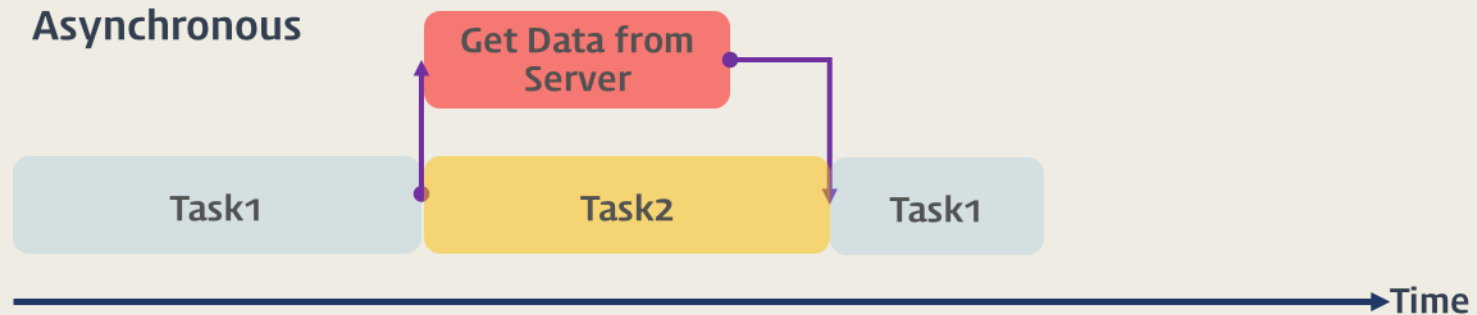
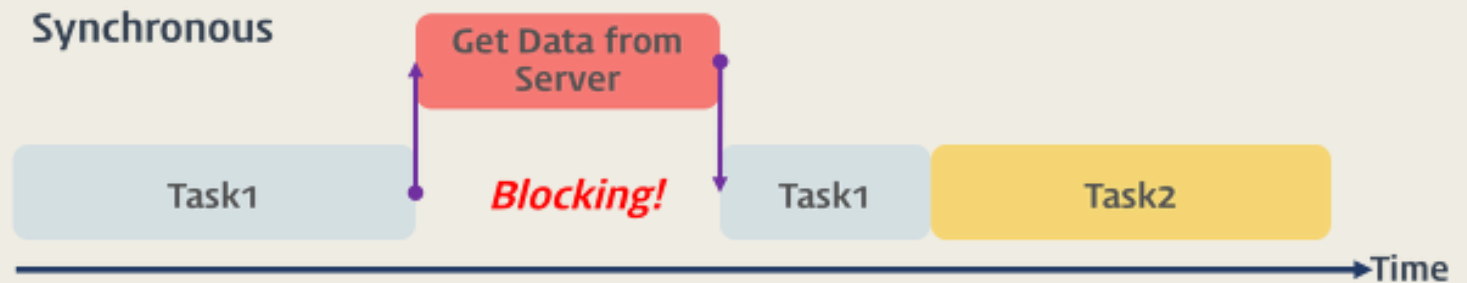
Useful Info : PIP

- Python 패키지 매니저
- Pip installs Packages 의 약자
- Python 2.7.9+ or Python 3.4+ 설치할 때 default로 같이 설치
- Ex. pip install 패키지명



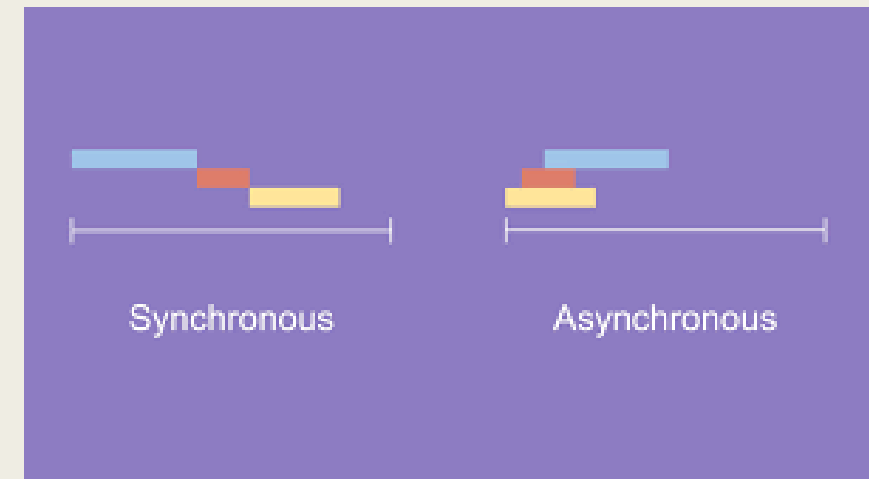
Useful Info: 비동기 프로그래밍

- Async (비동기) vs Sync (동기) ?



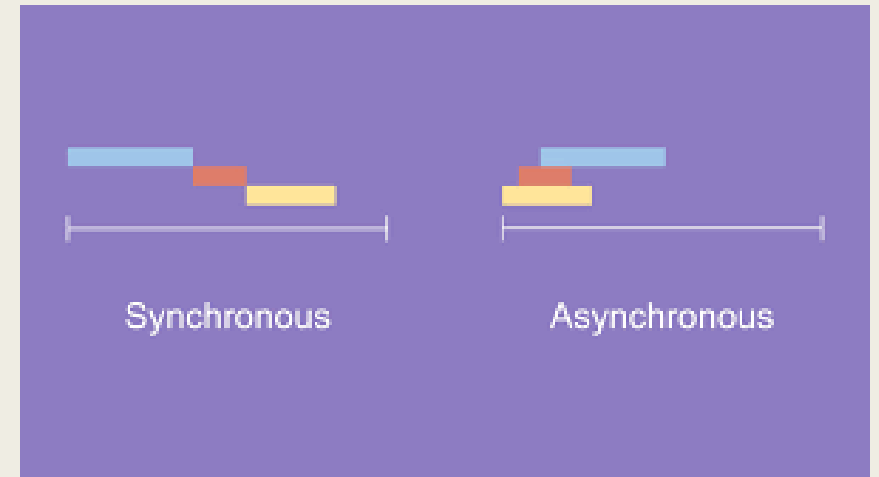
Useful Info: Sync

- 동기적인 처리 방식 : 직렬적으로 태스크를 완료한다.
- 장점
 - 설계가 매우 간단하다
 - 직관적이다
- 단점
 - 결과가 주어질 때까지 아무것도 못한다
 - 시간이 많이 소요가 된다.



Useful Info: Async

- 비동기 처리 방식: 병렬적으로 태스크를 수행한다
- 장점
 - 결과가 주어지는 시간동안 다른 작업을 할 수 있다
- 단점
 - 동기보다 복잡하다



Useful Info: Links

- [discord.py](https://github.com/Rapptz/discord.py) (<https://github.com/Rapptz/discord.py>)
- [Documents](https://discordpy.readthedocs.io/en/latest/) (<https://discordpy.readthedocs.io/en/latest/>)
- <https://github.com/hsmint/Python-DiscordBot-Whale>



Q & A

