

virtual environment report

우선 [pyenv](#)로 파이썬 버전관리를 하고 [pipenv](#)를 활용하여 가상환경을 만들었습니다.

```

repo/3-2-Python(2) git-main
> pipenv --python 3.9
Creating a virtualenv for this project...
Pipfile: /Users/gitp/gitp/ybigo.com/repo/3-2-Python(2)/Pipfile
Using Users/gitp/.pyenv/versions/3.9.15/bin/python3 (3.9.15) to create virtualenv...
Creating virtualenv /Users/gitp/.pyenv/versions/3.9.15/bin/python3.9.15-final.864 in 167ms
Created Python3Post(Host=Users/gitp/.local/share/virtualenvs/3-2-Python_2--5f790ktt, clear=False, no_vcs_ignore=False, global=False)
seeder FromAppData(download=False, pip=bundle, setuptools=bundle, wheel=bundle, via=copy, auto_detect=True, Library/Application Support/virtualenv)
added seed packages: pip==24.1, setuptools==70.1.1, wheel==0.43.0
activators BehAviActivator, CMakeActivator, FishActivator, NushellActivator, PowerShellActivator, PythonActivator
✓ Successfully created virtual environment!
Virtualenv location: /Users/gitp/.local/share/virtualenvs/3-2-Python_2--5f790ktt
Creating a PIP file for this project...

repo/3-2-Python(2) git-main
> pipenv run python 1-virtual-env.py

```

이런 식으로 버전에 따른 가상환경을 생성해 파일을 실행하고 가상환경을 삭제하는 것을 3번 반복했습니다.

아래는 마지막 실행결과입니다.

```
> pipenv run python 1-virtual-env.py
current: 3
dict:
3.12.4 (main, Jun 6 2024, 18:26:44) [Clang 15.0.0 (clang-1500.3.9.4)]
- /Users/gitp/git/ybigta_edu/rep0/3-2-Python(2)
- /opt/homebrew/opt/python@3.12.4/Frameworks/Python.framework/Versions/3.12/lib/python312.zip
- /opt/homebrew/Cellar/python@3.12.3/12.4/Frameworks/Python.framework/Versions/3.12/lib/python312
- /opt/homebrew/Cellar/python@3.12.3/12.4/Frameworks/Python.framework/Versions/3.12/lib/python3.12/lib-dynload
- /Users/gitp/.local/share/virtualenvs/3-2-Python_2-5f790xtt/lib/python3.12/site-packages
3.10.14 (main, Jul 17 2024, 22:31:22) [Clang 18.1.8]
- /Users/gitp/gitp/ybigta_edu/rep0/3-2-Python(2)
- /Users/gitp/.pyenv/versions/3.10.14/lib/python310.zip
- /Users/gitp/.pyenv/versions/3.10.14/lib/python310
- /Users/gitp/.pyenv/versions/3.10.14/lib/python3.10/lib-dynload
- /Users/gitp/.local/share/virtualenvs/3-2-Python_2-5f790xtt/lib/python3.10/site-packages
3.9.15 (main, Jul 9 2024, 15:51:38)
[Clang 18.1.8]
- /Users/gitp/gitp/ybigta_edu/rep0/3-2-Python(2)
- /Users/gitp/.pyenv/versions/3.9.15/lib/python39.zip
- /Users/gitp/.pyenv/versions/3.9.15/lib/python3.9
- /Users/gitp/.pyenv/versions/3.9.15/lib/python3.9/lib-dynload
- /Users/gitp/.local/share/virtualenvs/3-2-Python_2-5f790xtt/lib/python3.9/site-packages

good!
```

argparse

```
def create_parser() -> argparse.ArgumentParser:
    def str_to_bool(s: str) -> bool:
        if s == "True":
            return True
        elif s == "False":
            return False
        else:
            raise argparse.ArgumentTypeError("expected True or False")

    # 구현하세요!
    arg_parser: argparse.ArgumentParser =
argparse.ArgumentParser(description="foo parser")
arg_parser.add_argument("start", type=int)
```

```

arg_parser.add_argument("end", type=int)
arg_parser.add_argument("verbose", type=str_to_bool)
return arg_parser

```

start, end, verbose 순으로 argument를 전달받을 것을 기대하며 positional argument로 구현했습니다.

기존의 bool함수는 문자열이 존재하면 True, 존재하지 않으면 False를 반환하는 문제가 있었습니다. 그래서 "True", "False" 문자를 boolean형으로 바꿔주는 함수를 예외 처리를 하며 구현했습니다.

아래는 실행결과입니다.

```

▶ > python3 2-argparse.py 0 1 True
0 1 True

repo/3-2-Python(2) git:main
▶ > python3 2-argparse.py 0 1 False
0 1 False

repo/3-2-Python(2) git:main
▶ > python3 2-argparse.py 0 1 foo
usage: 2-argparse.py [-h] start end verbose
2-argparse.py: error: argument verbose: expected True or False

```

defaultdict

```

def __init__(self, default_producer: Callable[[], V]):
    super().__init__()
    self.default_producer: Callable[[], V] = default_producer

def __getitem__(self, key: K) -> V:
    try:
        return super().__getitem__(key)
    except KeyError:
        super().__setitem__(key, self.default_producer())
        return super().__getitem__(key)

```

defaultdict(lambda: [0, 1]) 이코드에서 힌트를 얻어서 초기값을 반환하는 함수를 인자로 받게 클래스를 구현했습니다. __setitem__ 은 호출할때 대입할 값을 인자로 주기 때문에 굳이 구현할 필요 없이 아직 생성되지 않은 key에 대해 초기값을 생성해주는 코드만 __getitem__에 넣어주었습니다.