

pre-commit

도입 이유

- 코드 포매터 pre-commit!
 - develop branch에 병합될 때 자동으로 점검 (GitLab CI)
 - 레포지토리를 관리하는 개발자들이 여럿이지만 마치 한 명이 개발한 것처럼 일관된 코드 컨벤션을 유지할 수 있음
 - 점검 사항 예시

```
(base) [base] zed ~/projects/lumini-algorithm-ubuntu feature/1-pre-commit
❸ git commit -m "[#1] inspect all files with precommit"
autoflake..... Passed
black..... Passed
isort (python)..... Passed
flake8..... Failed
  - hook id: flake8
  - exit code: 1

  APIs/Test_APIS.py:48:13: E741 ambiguous variable name 'l'
    l = []
    ^
  APIs/Test_APIS.py:59:41: F821 undefined name 'process'
    print(a, " memory -> ", process.memory_info().rss)
    ^
  APIs/Test_APIS.py:65:34: F821 undefined name 'process'
    print("main memory -> ", process.memory_info().rss)
    ^
  APIs/Test_APIS.py:68:35: F821 undefined name 'process'
    print("main 1memory -> ", process.memory_info().rss)
    ^
  APIs/Test_APIS.py:70:35: F821 undefined name 'process'
    print("main 2memory -> ", process.memory_info().rss)
    ^
  APIs/app.py:212:5: F824 `global luminiSkin` is unused: name is never assigned in scope
    global luminiSkin
    ^
  APIs/app.py:223:13: F841 local variable 'response' is assigned to but never used
    response = requests.post(
    ^
  APIs/app.py:337:9: F841 local variable 'response' is assigned to but never used
    response = requests.post(callback_url, headers=headers, data=json.dumps(err_json), timeout=600)
    ^
  AcnePigmentModule/models/unet_parts.py:103:24: E712 comparison to True should be 'if cond is True:' or 'if cond:'
    if self.onlyup == True:
    ^
  CartoonGeneratorModule/Test/Utils/CroppingPointUtils.py:197:1: F811 redefinition of unused 'get_cropping_points' from line 11
  def get_cropping_points(landmarks, input_size):
  ^
  CartoonGeneratorModule/Test/Utils/CroppingPointUtils.py:224:1: F811 redefinition of unused 'get_cropping_points_face' from line 38
  def get_cropping_points_face(landmarks, input_size):
  ^
  CartoonGeneratorModule/Test/facade/Test_PigmentEmojiGenerator.py:105:16: E711 comparison to None should be 'if cond is not None:'
    if ind != None:
    ^
  CartoonGeneratorModule/Test/facade/Test_WrinkleEmojiGenerator.py:125:16: E711 comparison to None should be 'if cond is not None:'
    if ind != None:
    ^
  CartoonGeneratorModule/facade/PigmentEmojiGenerator.py:105:16: E711 comparison to None should be 'if cond is not None:'
    if ind != None:
    ^
  CartoonGeneratorModule/facade/WrinkleEmojiGenerator.py:125:16: E711 comparison to None should be 'if cond is not None:'
    if ind != None:
    ^
  ClusterManager/cluster_manager.py:13:5: E722 do not use bare 'except'
  except:
  ^
  DarkcircleFlushModule/Test/DarkcircleFlushUnitTest.py:9:1: E402 module level import not at top of file
```

- 모호한 변수명
- 사용하지 않는 출처를 알 수 없는 객체 사용
- 참조되지 않는 전역 변수 선언
- 주소가 아닌 값 비교 에러
- 불필요한 함수 재정의
- except 이후 Exception 없음 (모면성 예외처리)

앞으로 미칠 영향

- 앞으로 git push 전에 pre-commit이 작동하고 문제가 있을 경우 포매터가 이를 고쳐줄 것임
 - 포매터가 직접 고칠 수 없는 경우 개발자가 수동으로 고쳐야함 (ex. defined but not used error)
 - 문제가 있는 경우 commit에서 넘어가지 않음. 그러므로 push 또한 하지 못함

- pull request의 gitlab action에도 달아놓고 이중으로 점검할 예정

사용 방법

- pre-commit 설치
 - pip install pre-commit
- 설치 pre-commit install
- 이후 동일(git add, git commit, git push)
 - commit에서 pre-commit이 오류들을 지적할 시 파일이 unstaged 상태가 됨
 - 일반적인 에러는 그냥 다시 add하고 commit하면 되지만 사람이 직접 확인하고 수정해야하는 경우 자동으로 fix되지 않음.
 - 수정하고 add commit push

사용 방법 detail

1.작업한 코드 변경사항을 unstage -> stage 추가한다.

```
git add {file_name}.py
```

2.메시지와 함께 수정 사항 detail을 commit 한다 (이 때 pre-commit trigger)

```
git commit -m "[#{issue_number}] add new method `get_face_landmark`"
```

case1: Success/Skipped

- 모든 포매터들의 검증을 마쳐 성공적으로 commit 된다.

case2: Failed

- 어디선가 검증에 실패해 commit되지 않고 다시 stage -> unstage로 내려간다.
- 파일을 수정하고 다시 git add {file_name} 해 주어야 한다.

대부분의 경우 pre-commit이 자동으로 고쳐주기에

```
git add .
git commit -m "fix pre-commit error"
```

위 명령을 반복하면 되지만 일부 문제들은 개발자가 직접 수정하여야 한다.

3.변경한 모든 파일들에 대한 검증이 모두 성공할때까지 1~2 번을 반복한다
모두 문제가 없으면 push

```
git push
```

예시 이미지 (1번 실패한 모습)

```
(3d-regist) [3d-regist] zed ~/projects/3d-regis/3d-registration > enhancement/1-cicd ± git add .
(3d-regist) [3d-regist] zed ~/projects/3d-regis/3d-registration > enhancement/1-cicd + git commit -m "[#1] fix unittest ci"
autoflake..... (no files to check) Skipped
black..... (no files to check) Skipped
isort (python)..... (no files to check) Skipped
flake8..... (no files to check) Skipped
trim trailing whitespace..... Failed
- hook id: trailing-whitespace
- exit code: 1
- files were modified by this hook

Fixing .gitlab-ci.yml

fix end of files..... Passed
check yaml..... Passed
(3d-regist) [x 3d-regist] zed ~/projects/3d-regis/3d-registration > enhancement/1-cicd ± git add .
(3d-regist) [3d-regist] zed ~/projects/3d-regis/3d-registration > enhancement/1-cicd + git commit -m "[#1] fix unittest ci"
autoflake..... (no files to check) Skipped
black..... (no files to check) Skipped
isort (python)..... (no files to check) Skipped
flake8..... (no files to check) Skipped
trim trailing whitespace..... Passed
fix end of files..... Passed
check yaml..... Passed
[enhancement/1-cicd f011796] [#1] fix unittest ci
  1 file changed, 33 insertions(+), 14 deletions(-)
(3d-regist) [3d-regist] zed ~/projects/3d-regis/3d-registration > enhancement/1-cicd > git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 10 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 854 bytes | 854.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: View merge request for enhancement/1-cicd:
remote:   https://gitlab.com/lululab1/aicenter/3d-registration/-/merge_requests/3
remote:
To https://gitlab.com/lululab1/aicenter/3d-registration.git
  6c00672..f011796 enhancement/1-cicd -> enhancement/1-cicd
```