**데이터 베이스**

1. DB 설정 및 테이블 설정

* DB 설정

|  |  |
| --- | --- |
| DB엔진 버전 | 10.3.11 |
| 호스트 | onad.cbjjamtlar2t.ap-northeast-2.rds.amazonaws.com |
| 포트 | 3306 |
| 데이터베이스 명 | onad |
| DB user | onad |
| DB password | \*\*\*\*\*\*\*\*\*\* |
| Charset | utf8mb4\_general\_ci |
| Log flag | False |

1. 테이블 설정

* 트위치 데이터 테이블

TwitchChat(Base):

"""

트위치 스트리밍의 채팅데이터를 적재하는 테이블

-> chatty 로그를 하루 단위로 저장

chat\_id: 채팅의 번호

streamer\_name : 스트리머의 이름

broad\_date : 해당 채팅의 방송을 한 날짜

chatterer : 채팅을 친 시청자의 이름

chat\_time : 채팅을 친 시간정보

chat\_contents : 채팅정보

"""

\_\_tablename\_\_ = 'twitch\_chat'

chat\_id = Column(Integer, primary\_key=True, autoincrement=True)

streamer\_name = Column(String(50), unique=False)

broad\_date = Column(String(50), unique=False)

chatterer = Column(String(50), unique=False)

chat\_time = Column(String(50), unique=False)

chat\_contents = Column(Text, unique=False)

TwitchStream(Base):

"""

streamer\_id: twitch 스트리머의 고유 ID

stream\_id: twitch 생방송의 고유 ID(this changes every stream )

streamer\_name: 해당 스트리머 이름

broad\_date: 해당 스트리밍의 방송시작날짜

"""

\_\_tablename\_\_ = 'twitch\_stream'

code = Column(Integer, primary\_key=True, autoincrement=True)

stream\_id = Column(String(50), unique=False)

streamer\_id = Column(String(50), unique=False)

streamer\_name = Column(String(50), unique=False)

broad\_date = Column(String(50), unique=False)

TwitchStreamDetail(Base):

"""

트위치 스트리밍의 세부정보를 담기 위한 테이블

stream\_id: twitch 생방송의 고유 ID( this changes every stream )

viewer: 시청자수

title: twitch 생방송의 제목

game\_id: 진행중인 게임의 고유 ID

date: 시간 정보

"""

\_\_tablename\_\_ = 'twitch\_stream\_detail'

code = Column(Integer, primary\_key=True, autoincrement=True)

stream\_id = Column(String(50), unique=False)

viewer = Column(Integer, unique=False)

title = Column(String(150), unique=False)

game\_id = Column(String(50), unique=False)

time = Column(TIMESTAMP, default=func.now())

TwitchChannel(Base):

"""

트위치 채널정보를 담아놓는 테이블

streamer\_id: twitch 스트리머의 고유 ID

streamer\_name: 스트리머의 이름(닉네임)

logo: 스트리머 로고의 주소

homepage: 스트리머 방송국 홈페이지

"""

\_\_tablename\_\_ = "twitch\_channel"

code = Column(Integer, primary\_key=True, autoincrement=True)

streamer\_id = Column(String(50), unique=False)

streamer\_name = Column(String(50), unique=False)

logo = Column(Text, unique=False)

homepage = Column(Text, unique=False)

TwitchChannelDetail(Base):

"""

트위치 채널의 세부정보를 담아놓는 테이블로

잦은 업데이트가 필요한 테이블

streamer\_id: twitch 스트리머의 고유 ID

date: 시간 정보

follower: 팔로워의 수

subscriber: 구독자의 수

"""

\_\_tablename\_\_ = "twitch\_channel\_detail"

code = Column(Integer, primary\_key=True)

streamer\_id = Column(String(50), unique=False)

date = Column(TIMESTAMP, default=func.now())

follower = Column(Integer, unique=False)

viewer = Column(Integer, unique=False)

TwitchGame(Base):

"""

트위치에서 제공하는 게임정보를 담아놓는 테이블

game\_id: 트위치에서 설정한 게임의 고유 ID

game\_name: 게임 이름

"""

\_\_tablename\_\_ = "twitch\_game"

code = Column(Integer, primary\_key=True, autoincrement=True)

game\_id = Column(String(50), unique=False)

game\_name = Column(String(100), unique=False)

date = Column(TIMESTAMP(100), default=func.now(), unique=False)

TwitchGameDetail(Base):

"""

twitch\_game 테이블의 세부사항을 담아놓는 테이블

잦은 업데이트가 필요한 테이블

game\_id: 트위치에서 설정한 게임의 고유 ID

date: 현재 날짜

all\_viewer: 이 게임을 시청하는 총 시청자

stream\_this\_game: 이 게임을 방송중인 스트리밍의 수

"""

\_\_tablename\_\_ = "twitch\_game\_detail"

code = Column(Integer, primary\_key=True, autoincrement=True)

game\_id = Column(String(50),unique=False)

date = Column(TIMESTAMP, default=func.now())

all\_viewer = Column(Integer, unique=False)

stream\_this\_game = Column(Integer, unique=False)

TwitchFollowing(Base):

"""

twitch\_following 테이블의 세부사항을 담아놓는 테이블

user\_id: 유저 고유 ID

following\_streamer: 팔로우하는 스트리머

streamer\_name: 스트리머 이름

followed\_at: 팔로우 한 날짜

"""

\_\_tablename\_\_ = "twitch\_following"

code = Column(Integer, primary\_key=True, autoincrement=True)

user\_id = Column(String(50),unique=False)

following\_streamer = Column(String(50),unique=False)

streamer\_name = Column(String(50),unique=False)

followed\_at = Column(String(50),unique=False)

TwitchClip(Base):

"""

트위치 클립 데이터를 담아놓는 테이블

streamer\_id: 클립영상의 방송인의 고유 ID

clip\_id: 클립영상의 고유 id

user\_id: 클립 생성자의 고유 ID

created\_at: 클립이 생성된 날짜및 시간

title: 클립의 제목

url: 클립영상의 주소

viewer\_count: 클립 조회수

thumbnail: 썸네일 주소

"""

\_\_tablename\_\_ = "twitch\_clip"

code = Column(Integer, primary\_key=True, autoincrement=True)

streamer\_id = Column(String(50), unique=False)

clip\_id = Column(String(150), unique=False)

user\_id = Column(String(50), unique=False)

created\_at = Column(String(50), unique=False)

title = Column(String(100), unique=False)

url = Column(Text, unique=False)

viewer\_count = Column(String(50), unique=False)

thumbnail = Column(Text, unique=False)

* 유튜브 데이터 테이블

YoutubeChannel(Base):

"""

유튜브 채널정보를 담기위한 테이블

channel\_id: 유튜브에서 설정한 고유 ID

channel\_name: 채널 이름

description: 채널 설명

published\_at: 채널 생성일

thumbnail: 채널 썸네일

keyword: 채널 검색 키워드

recommend\_channels: 추천하는 채널

"""

\_\_tablename\_\_ = 'youtube\_channel'

code = Column(Integer, primary\_key=True, autoincrement=True)

channel\_id = Column(String(50), unique=True)

channel\_name = Column(String(50), unique=False)

description = Column(Text, unique=False)

published\_at = Column(String(50), unique=False)

thumbnail = Column(String(50), unique=False)

keyword = Column(String(150), unique=False)

recommend\_channels = Column(Text, unique=False)

YoutubeChannelDetail(Base):

"""

유튜브 채널의 세부 정보를 담기위한 테이블

자주 업데이트 되는 항목들 모음

channel\_id: 유튜브에서 설정한 고유 ID

subscribe\_cnt: 채널 구독자 수

hit\_cnt: 채널 조회 수

total\_video\_cnt: 채널 영상 수

"""

\_\_tablename\_\_ = 'youtube\_channel\_detail'

code = Column(Integer, primary\_key=True)

channel\_id = Column(String(50), unique=False)

subscribe\_cnt = Column(Integer, unique=False)

hit\_cnt = Column(Integer, unique=False)

total\_video\_cnt = Column(Integer, unique=False)

YoutubeVideo(Base):

"""

각 채널의 동영상들의 정보를 저장하는 테이블

지속적인 업데이트(갈아끼우기) 필요한 테이블

channel\_id: 유튜브에서 설정한 채널의 고유 ID

video\_id: 영상의 고유 ID

title: 영상 제목

description: 영상 설명

upload\_date: 영상 게시 날짜

tag: 영상 태그

thumbnail: 썸네일 주소

is\_live: 업로드된 라이브방송인지, 일반 영상인지의 여부

view\_cnt: 영상 조회수

like\_cnt: 좋아요 수

hate\_cnt: 싫어요 수

reple\_cnt: 댓글 수

category: 카테고리

is\_live: 라이브방송이었는지 아닌지

"""

\_\_tablename\_\_ = 'youtube\_video'

code = Column(Integer, primary\_key=True)

channel\_id = Column(String(100), unique=False)

video\_id = Column(String(100), unique=False)

title = Column(String(100), unique=False)

description = Column(Text, unique=False)

upload\_date = Column(String(50), unique=False)

tag = Column(Text, unique=False)

category = Column(String(50), unique=False)

thumbnail = Column(Text, unique=False)

view\_cnt = Column(Integer, unique=False)

like\_cnt = Column(Integer, unique=False)

hate\_cnt = Column(Integer, unique=False)

reple\_cnt = Column(Integer, unique=False)

is\_live = Column(String(20), unique=False)

YoutubeChat(Base):

"""

유튜브 라이브 동영상의 채팅데이터를 저장하는 테이블

chat\_id: 채팅의 고유 번호

video\_id: 영상의 고유 ID

chat\_time: 채팅을 친 시간

chatterer: 채팅을 친 시청자의 이름

chat\_contents: 채팅 내용

broad\_date: 방송 날짜

"""

\_\_tablename\_\_ = 'youtube\_chat'

code = Column(Integer, autoincrement=True, primary\_key=True)

chat\_id = Column(String(50), unique=False)

video\_id = Column(String(100), unique=False)

chat\_time = Column(String(50), unique=False)

chatterer = Column(String(50), unique=False)

chat\_contents = Column(Text, unique=False)

broad\_date = Column(String(50), unique=False)

YoutubeReple(Base):

"""

유튜브 영상의 댓글정보를 저장하는 테이블

reple\_id: 댓글 작성자의 고유 채널 ID

video\_id: 영상의 고유 ID

upload\_date: 댓글을 단 시간

author\_name: 댓글을 단 시청자의 이름

reple\_contents: 댓글 내용

like\_cnt: 댓글 좋아요 수

"""

\_\_tablename\_\_ = 'youtube\_reple'

code = Column(Integer, primary\_key=True, unique=False)

reple\_id = Column(String(100), unique=False)

video\_id = Column(String(100), unique=False)

upload\_date = Column(String(50), unique=False)

author\_name = Column(String(50), unique=False)

reple\_contents = Column(Text, unique=False)

like\_cnt = Column(Integer, unique=False)

YoutubeSubscription(Base):

"""

댓글 남긴 인원의 구독 정보를 저장하는 테이블

code: 구독정보의 번호

video\_id: 영상ID

replier\_id: 댓글 시청자의 고유 채널 ID

subscription: 구독채널ID

"""

\_\_tablename\_\_ = 'youtube\_subscription'

code = Column(Integer, primary\_key=True, unique=False)

video\_id = Column(String(100), unique=False)

replier\_id = Column(String(50), unique=False)

subscription = Column(String(50), unique=False)

* 회원정보 관리 테이블

Creater(Base):

"""

크리에이터 회원의 정보를 담는 테이블

code: 회원 번호

reg\_date: 가입 일시

flatform: 방송 플랫폼 정보

creater\_id: (youtube면 channel\_id twitch면 streamer\_id)

user\_id: onad 가입한 아이디

user\_pw: onad 가입한 비밀번호

user\_name: onad 가입한 이름

nickname: 닉네임(creater 활동명)

phone: 전화번호

email: 이메일주소

homepage: 홈페이지 주소

broad\_program: 방송 송출 프로그램 명

"""

\_\_tablename\_\_ = "creater"

code = Column(Integer, primary\_key=True, autoincrement=True)

reg\_date = Column(TIMESTAMP, default=func.now())

flatform = Column(String(50), unique=False)

creater\_id = Column(String(50), unique=False)

user\_id = Column(String(20), unique=False)

user\_pw = Column(String(15), unique=False)

user\_name = Column(String(30), unique=False)

nickname = Column(String(30), unique=False)

phone = Column(String(30), unique=False)

email = Column(String(100), unique=False)

homepage = Column(Text, unique=False)

broad\_program = Column(String(30), unique=False)

Advertiser(Base):

"""

광고주 회원의 정보를 담는 테이블

code: 회원 번호

reg\_date: 가입 일시

user\_id: onad 가입한 아이디

user\_pw: onad 가입한 비밀번호

user\_name: onad 가입한 이름 (담당자 명)

phone: 담당자 전화번호

email: 담당자 이메일주소

corporation\_name: 소속기업 명

corporation\_code: 소속기업의 코드

\* 다른 테이블과는 다르게 dict형태의 데이터를 집어넣어야 한다.

{'user\_id': user\_pw, 'user\_pw': user\_pw, ...}

"""

\_\_tablename\_\_ = "advertiser"

code = Column(Integer, primary\_key=True, autoincrement=True)

reg\_date = Column(TIMESTAMP, default=func.now())

creater\_id = Column(String(50), unique=False)

user\_id = Column(String(20), unique=False)

user\_pw = Column(String(15), unique=False)

user\_name = Column(String(30), unique=False)

nickname = Column(String(30), unique=False)

phone = Column(String(30), unique=False)

email = Column(String(100), unique=False)

corporation\_name = Column(String(30), unique=False)

corporation\_code = Column(String(30), unique=False)

Corporation(Base):

"""

기업 정보를 담는 테이블

code: 가입된 기업 번호

corporation\_code: 소속기업의 코드

corporation\_name: 소속기업 명

nation: 소속 국가

chief: 대표자 명

product: 업종

prior\_product: 주요 상품 군

homepage: 회사홈페이지

corporation\_num: 사업자 등록번호

\* 다른 테이블과는 다르게 dict형태의 데이터를 집어넣어야 한다.

{'user\_id': user\_pw, 'user\_pw': user\_pw, ...}

"""

\_\_tablename\_\_ = "corporation"

code = Column(Integer, primary\_key=True, autoincrement=True)

corporation\_code = Column(String(50), unique=False)

corporation\_name = Column(String(50), unique=False)

nation = Column(String(30), unique=False)

chief = Column(String(30), unique=False)

product = Column(String(30), unique=False)

prior\_product = Column(String(30), unique=False)

homepage = Column(String(100), unique=False)

corporation\_num = Column(String(50), unique=False)

* 모델 버전 관리 테이블

ModelVersion(Base):

"""

감성분석 모델의 버전정보를 담고있는 테이블

version: 버전 정보

date: 업데이트 날짜

file\_name: 모델 파일명

f1\_score: f1-score 점수

accuracy: 정확도 점수

comment: 코멘트

"""

\_\_tablename\_\_ = "model\_version"

version = Column(Integer, primary\_key=True)

date = Column(TIMESTAMP, default=func.now())

file\_name = Column(String(100), unique=False)

f1\_score = Column(Float, unique=False)

accuracy = Column(Float, unique=False)

comment = Column(Text, unique=False)