to do

1. major
   1. account migration
   2. log
   3. 테스트
   4. rollback 로직 고도화
   5. 잉여 자원 삭제 알고리즘
      1. db 와 ktcloud 동기화
         1. ktcloud 조회 api 로 id list 확인하여 db 와 비교
      2. db 에서 잉여자원 확인하여 삭제
2. minor
   1. input, output 수정
      1. 연속 메소드 활용
   2. 토큰 생성 메소드
   3. tier creation, deletion
   4. CONF 파일 상대경로로 읽기
   5. retry
   6. 소스 정리
      1. 불필요한 부분 제거
      2. 주석 달기

memo

* 1. 로그
     1. <https://engkimbs.tistory.com/861>
     2. <https://velog.io/@zzarbttoo/%EB%A1%9C%EA%B7%B8log4j2%EB%A5%BC-%EC%9D%B4%EC%9A%A9%ED%95%B4-%ED%8A%B9%EC%A0%95-%ED%8C%8C%EC%9D%BC%EC%97%90-%EB%A1%9C%EA%B7%B8-%EA%B8%B0%EB%A1%9D%EC%9D%84-%EB%AA%A8%EC%95%84%EB%B3%B4%EC%95%98%EB%8B%A4>
     3. <https://www.digitalocean.com/community/tutorials/log4j2-example-tutorial-configuration-levels-appenders#log4j2-levels>

1. 서버 응답 경우의 수
   1. 성공
   2. 처리중
      1. 클라이언트 대기 하기
   3. 실패
      1. 클라이언트 retry 하기
2. status code 로 성공실패 여부를 확인하는것 보단 응답 메시지를 파싱해서 판단해야 할듯