강의 교재

https://drive.google.com/drive/folders/1zjTtqnGOAdbxMga7nWi8uLElBnnuUATa

강의 자료

https://mybox.naver.com/share/list?shareKey=qNiQyfILf0d0SVD5vQikmKl6T3DvuLwt53Ym0utoFjQD

- 김윤수 강사님

- 개발자

- 클라우드에 개발자

- 보안/품질 관리

- 책집필: ISMS, 모의 해킹/취약점, 보안 테스트

- 01086384839

- kchris000@gmail.com

! SW테스트와 품질\_ 김윤수

! 단위 테스트 - 인프라/QA 부서와의 협업 및 자동화

================1일차================

- 셀프 피드백

[TDD의 목적]

* 소프트웨어 개발(품질, 테스트)를 위해
* TDD, Test 자동화

[강의 수행]

* 파이썬 기반으로 테스트
* 파이Test
* 코드는 도구와 제품의 방향성에 따라 달라질 수 있다.
* 이미지 매칭 기반
* 스크립트를 통한 자동화

[TDD가 유용할 때]

* 요구사항이 급변할 때

[개발 고민 요소]

* 개발하고, 테스트한 그 코드가 서버에 어떻게 배포되었음을 보장하지?  
  🡺 기능단위를 도커로 묶어서 이미지 배포
* 클라우드 인프라와 운영서버 인프라가 같다는 것을 어떻게 보장할 거야?  
  🡺 테라폼 도입
* 로그 파일이 너무 많이 쌓여서 죽어버렸는데 분석도구는 어떻게 할 거야?  
  🡺 문제 생기면, 알려주는 체제 도입
* MSA, Cloud, Docker 이 상황에서 품질확보 및 언제든 서비스 할 수 있다는 환경을 어떻게 보장할 것인가?  
  🡺

[개발보안진단 도구-보안적인 이슈가 있는지 확인]

* sonar-scanner

! 전체적으로 요구사항을 어떻게 충족하고, 안정성을 어떻게 확보하냐

[테스트 종류]

* 단위 테스트
* 시스템 테스트: 스트레스 테스트(성능 테스트)
* 모듈과 모듈간 테스트

! 파이썬, 3.11버전부터 엔진이 바뀌어 성능이 좋아짐

! SikuliX – 이미지 매칭 테스트

! 지향점

* 요구사항 분석
* 결과에 대한 검증 방안에 대해 고민
* 상황과 환경이 바뀌었을 때 대응 🡺 컨테이너 단위로 배포 🡺 배포하기 전 테스트 수행

첫날 순서

1. 인프라 설정
2. 협업 요소
3. 소프트웨어 공학(요구사항, 개발 방향, 생산성-품질확보)

\* 환경 변수: 운영체제가 참조하는 글로벌 변수

! 환경변수 역할: 간접 참조

ex)

(win)

SET MYENV=HELLO

echo %MYENV%

(linux)

export, $MYENV

\* 환경변수 등록

- 윈도우버튼+R

- sysdm.cplr

! 설정한 다음 재부팅을 해줘야 함

JDK 환경 변수 잡아주기

‘JAVA\_HOME, PATH’