

# AI활용프로그래밍

## Week 14. 개인 프로젝트 발표

발표 운영 · 제출물 · 피드백

## 이번 주 목표

개인별 창의적 프로그램 과제를 발표하고 피드백을 받는다.

프로그램의 문제 정의/기능/데이터/AI 활용 내역을 명확히 설명한다.

기말(Week15) 대비 제출물(코드/README/기록)을 정리한다.

# 오늘의 진행(예시)

- 발표 순서 안내(랜덤/번호/자원)
- 발표(각 3~5분) + 질의응답(1~2분)
- 피드백 기록(개선점 2개 이상)
- 마무리: 제출물 체크리스트 & 기말 안내

# 제출물

## ■ < 보고서 >

- 사용 데이터셋
- 해결 문제 정의 : 알아 내고자 하는 문제를 정의함 (이유 포함)
- Readme.md(설치/실행/사용법/예시 출력)
- LLM 사용 기록 (prompt log): 프롬프트/요약/반영 내용
- 실행 결과 (결과 화면 캡처 이미지)

## ■ < 소스코드(.py) 파일 >

- 소스코드에 대한 설명 (매 라인 주석처리)

## ■ < 발표 자료 >

- 보고서 내용 요약 설명
- 시연 (짧은 시연 영상으로 대체 가능)

# 발표에서 꼭 보여줄 것(핵심 5)

- 1) 문제/목표: 무엇을 해결하려 했나?
- 2) 주요 기능 2~3개(데모)
- 3) 핵심 코드 구조(함수/모듈/자료구조)
- 4) 테스트/엣지 케이스(어떤 입력에서 안정적인가)
- 5) AI(LLM) 활용 내역(어디에, 왜 썼나)

## 데모 준비 팁

- 데모용 입력/데이터는 미리 준비(실패 방지)
- 출력은 보기 좋게(메뉴, 안내 문구, 포맷)
- 에러가 나도 당황하지 말고: 의도/원인/개선 계획 설명
- 가능하면 짧은 데모 시나리오를 1장으로 적어두기

## 평가 루브릭(예시)

기능 완성도(요구 기능 동작) 40%

창의성/문제 해결 아이디어 20%

코드 품질(가독성/구조/주석) 20%

테스트/예외 처리 10%

AI 활용 기록과 설명 10%

※ 실제 배점은 수업 공지 기준

## 발표 시간 관리(꿀팁)

3~5분 발표 기준: 설명 60초 + 데모 60초 + 정리 30초

데모는 '핵심 기능 1개'만 확실히 보여줘도 충분

기능을 다 보여주려다 시간 초과/오류가 나기 쉬움

슬라이드가 없어도 '구조'만 지키면 전달력 ↑

# 발표 슬라이드(또는 말) 구조 추천

- 1장: 프로젝트 제목 + 한 줄 소개
- 2장: 기능 목록(체크박스 3개 정도)
- 3장: 데모 시나리오(입력→출력)
- 4장: 기술 포인트(자료구조/라이브러리/알고리즘)
- 5장: AI 활용 내역 + 배운 점/개선점

# README 템플릿(간단)

# 프로젝트 이름

- 한 줄 설명

## 실행 방법

- python main.py

## 사용법

- 메뉴/입력 예시

## 기능

- [ ] 기능1
- [ ] 기능2

## AI 활용 내역

- 사용한 프롬프트 1~3개 요약
- 반영한 수정 사항

# AI 활용 내역 작성법(추천 형식)

프롬프트: 내가 AI에게 무엇을 요청했나?

요약: AI 답변의 핵심은 무엇이었나?

반영: 내 코드/설계에 어떻게 적용했나?

검증: 실행/테스트 결과는 어땠나?

주의: '그대로 복붙'이 아니라 '이해 후 반영'

# AI 활용 표기(윤리) 한 줄 정리

AI가 만든 코드를 그대로 제출하면 '내 실력'이 드러나지 않는다  
프롬프트/반영 내용/검증 결과를 기록하면 학습이 증명된다  
외부 코드/자료를 참고했다면 출처를 남기는 습관  
목표: AI와 협업하되, 최종 책임은 작성자에게

# 발표/데모 실패를 줄이는 체크

환경 고정: 실행 폴더/가상환경/패키지 확인

입력은 최소 2세트 준비(정상/엣지)

랜덤 요소가 있으면 seed 고정(선택)

시간 초과 방지: 데모는 30~60초로 끊기

## 코드 품질 체크리스트(최종 정리용)

함수/변수명이 의미를 드러내는가?

중복 코드가 있는가? (함수로 묶을 수 있나)

입력값 검증/예외 처리 1~2개라도 있는가?

출력 메시지가 사용자 친화적인가?

README의 실행 방법이 '한 번에' 따라 되는가?

## 프로젝트 아이디어 예시(참고)

텍스트 기반 게임(퀴즈/반응속도/미니 RPG)

To-do/가계부/성적 관리(파일 저장 포함)

간단 데이터 분석 리포트(CSV→필터→그래프→요약)

웹 크롤링은 환경/정책 이슈가 있으니 수업 지침 준수

내가 자주 하는 일을 자동화하는 스크립트

# With AI 실습 1: 아이디어 업그레이드

## 프롬프트 템플릿

- 1 역할: 너는 프로젝트 멘토야.
- 2 목표: 아래 프로젝트 아이디어를
- 3 - 더 구체적인 요구사항
- 4 - 추가 기능 3개(난이도 낮음→중간→도전)
- 5 - 필요한 자료구조/모듈
- 6 로 확장해줘.
- 7 내 아이디어:
- 8 [여기에 적기]

## 핵심 포인트

아이디어를 '요구사항'으로 바꾸는  
과정이 중요  
추가 기능은 난이도 별로 단계화  
자료구조/모듈을 같이 받으면 구현  
이 쉬움

# With AI 실습 2: 테스트/엣지 케이스 만들기

## 프롬프트 템플릿

- 1 역할: 너는 QA 엔지니어야.
- 2 목표: 아래 프로그램의
- 3 - 엣지 케이스 8개
- 4 - 테스트 시나리오 8개(입력→기대 결과)
- 5 를 만들어줘.
- 6 프로그램 설명:
- 7 [여기에 기능/입력 형식]

## 핵심 포인트

발표에서 '테스트' 언급하면 완성도  
↑

엣지 케이스를 잡으면 오류가 줄어  
듦

테스트는 보고서/README에도 사  
용

# With AI 실습 3: README/설명문 작성

## 프롬프트 템플릿

- 1 역할: 너는 기술 문서 작성자야.
- 2 목표: 아래 프로젝트를 위한 README 초안을 작성해줘.
- 3 포함: 실행 방법, 사용법, 기능, 예시 출력, AI 활용 내역 섹션
- 4 자료:
- 5 - 프로젝트 설명: ...
- 6 - 실행 명령: python ...
- 7 - 예시 입력/출력: ...

## 핵심 포인트

- 문서는 '사용자 관점'이 중요
- 예시 입력/출력을 넣으면 이해가 빠름
- 발표 자료로도 바로 활용 가능

# With AI 실습 4: 발표 스크립트(1분 버전)

## 프롬프트 템플릿

- 1 역할: 너는 발표 코치야.
- 2 목표: 아래 프로젝트를 1분 발표로 요약해줘.
- 3 구성: 문제→핵심 기능→데모 포인트→배운 점
- 4 내 프로젝트 요약:
- 5 [여기에 5줄로 적기]

## 핵심 포인트

짧은 버전이 있으면 발표가 안정적  
데모 중 문제가 생겨도 말로 설명  
가능  
핵심은 '왜 만들었는가'

# 동료 피드백 가이드(서로에게)

좋았던 점 1가지(구체적으로)

개선하면 더 좋아질 점 1~2가지

버그/엣지 케이스 제안 1가지

코드 가독성/구조 제안 1가지

AI 활용 기록이 명확했는지 확인

## 발표자 체크리스트(마지막 5분)

- 프로그램이 처음부터 끝까지 실행되는지 확인
- 데모 입력/데이터 파일이 폴더에 있는지 확인
- README에 실행 방법이 있는지 확인
- LLM 사용 기록 1개 이상 정리
- 시간(3~5분) 안에 끝나는지 리허설

## 실습: 10분 리허설(수업 중)

- 1) 1분 설명(문제/기능) 말로 해보기
  - 2) 데모 30초 진행
  - 3) 질문 1개 예상해 답해보기
  - 4) 개선점 1개 메모
- 시간이 남으면 코드 구조(함수/자료구조) 한 번 더 정리

# 자주 받는 질문(예상 Q&A)

왜 이 주제를 선택했나요?

가장 어려웠던 버그는 무엇이었나요?

AI를 어디에 활용했고, 무엇을 수정했나요?

테스트 케이스는 어떻게 만들었나요?

다음 버전에서 추가하고 싶은 기능은?

# 과제/제출 안내(정리)

발표 후: 피드백을 반영한 최종 버전으로 정리  
폴더 구조/파일명 정리 + 불필요 파일 제거  
README와 LLM 사용 기록을 '최종 제출물' 수준으로  
기말(Week15)에는 평가/정리 일정이 있으니 공지 확인

## 요약 & 다음 주 예고

발표는 '기능'뿐 아니라 '과정/설명/검증'이 중요하다.

README와 AI 활용 기록은 프로젝트 완성도를 결정한다.

피드백을 반영해 최종 버전을 다듬자.

다음 주: 기말 시험(또는 최종 평가) 진행

# 데이터 셋과 주요 주제 (샘플)

## ■ 타이타닉(Titanic) 데이터셋

- 생존률이 가장 높은 그룹 찾기
- 나이대별 생존률 분석하기
- 간단한 생존 예측 모델 만들기

## ■ 넷플릭스(Netflix) 데이터셋

- 연도별 영화 증가 추세 분석
- 영화 장르와 TV 쇼 등의 비율 분석
- 장르와 국가별 인기도 상관관계 분석

## ■ 50년월드컵 데이터셋

- 개최국과 승률과의 상관관계 분석
- 현대 축구의 흐름 변화 찾기
- 강팀과 약팀을 구분하기
- 관중수에 영향을 주는 요인을 찾기