

날짜 (Day 1~28)	학습 목표	학습/실습 내용	실습 과제 또는 산출물	예 상 소 요 시 간
Day 1	에듀테크 1인 강의 자동화 시스템 개요 파악 및 환경 설정	프로젝트 목표와 주요 도구(AppSheet, n8n, OpenAI API 등)의 역할을 이해하고, 필요한 계정/API 키를 발급 및 설정한다. n8n 실행 환경(클라우드 또는 로컬)을 준비하여 개발에 착수할 기반을 마련한다.	계정 및 API 키 준비 완료, 전체 시스템 아키텍처 개요 정리(다이어그램 또는 설명)	2 시 간
Day 2	AppSheet 기본 학습 및 간단한 앱 생성	Google Sheets와 연동된 AppSheet 앱을 신규 생성하여 기본 인터페이스를 익힌다. 강의 정보 저장용 구글 시트를 준비하고 AppSheet로 연결한 후, 기본 입력 폼을 구성하여 샘플 데이터를 입력해본다.	Google Sheet 기반 강의 정보 입력용 AppSheet 앱 초안 완성 (샘플 데이터 입력 확인)	2 시 간
Day 3	AppSheet 앱 필드 확장 및 폼 개선	OpenAI 콘텐츠 생성을 위해 필요한 강의 정보 항목(주제, 대상, 목표 등)을 파악하여 구글 시트 및 AppSheet 폼에 해당 필드를 추가한다. 추가한 필드에 대해 드롭다운/유효성 검사 등을 설정하고, 예시 강의 정보를 입력하여 앱 동작을 재검증한다.	강의 계획 생성을 위한 필드가 모두 갖춰진 AppSheet 폼 완성 (예시 데이터로 테스트 완료)	2 시 간
Day 4	n8n 워크플로우 기본 사용법 익히기	n8n 에디터 UI와 노드 연결 개념을 학습하고, 간단한 워크플로우를 만들어본다. 수동 트리거와 함수 노드를 사용해 "Hello World" 등의 출력을 테스트하고, Webhook 트리거 노드를 생성해 간단한 요청을 수신해본다.	n8n 기본 워크플로우 (수동 트리거→간단 출력) 구현 및 Webhook 수신 테스트 완료	2 시 간
Day 5	n8n에서 Google Sheets 연동 실습	n8n 워크플로우에 Google Sheets 노드를 추가하여 AppSheet 입력 데이터를 읽어오는 연습을 한다. Google API 인증을 설정하고 구글 시트의 최신 레코드를 불러와보며, 필요하다면 정기 실행 등 트리거 방식을 검토한다.	n8n에서 Google Sheets 연동 성공 (시트 데이터 읽기 워크플로우 구현 및 테스트)	2 시 간
Day 6	AppSheet와 n8n 통합 (웹훅 연동)	AppSheet의 Automation 기능으로 새로운 입력 발생 시 n8n 웹훅을 호출하도록 설정한다. n8n 쪽에서는 Webhook 트리거로 데이터를 받고 다음 처리를 할 수 있게 워크플로우를 구성한다. AppSheet 앱에서 폼을 제출하여 n8n이 데이터를 수신하는지 통합 테스트를 진행한다.	AppSheet→n8n 웹훅 연동 완료 (폼 제출 시 n8n 워크플로우에서 데이터 수신 확인)	2 시 간

날짜 (Day 1~28)	학습 목표	학습/실습 내용	실습 과제 또는 산출물	예 상 소 요 시 간
Day 7	1주차 리캡 및 포트폴리오 정리 (일요일)	1주 동안 학습한 AppSheet 기본 기능과 n8n 연동 내용을 복습하고 정리한다. 현재까지 만든 AppSheet 앱과 n8n 워크플로우를 점검하여 개선할 사항을 기록하고, 주간 학습 내용을 포트폴리오(블로그 등)에 캡처 이미지와 함께 정리한다.	1주차 학습 요약 및 결과물 포트폴리오 기록 (스크린샷 및 설명 정리)	2 시 간
Day 8	OpenAI API 이해 및 사용 준비	OpenAI의 GPT API 엔드포인트와 요청/응답 구조를 학습한다. 간단한 프롬프트로 OpenAI API를 호출해보면서 (예: curl 또는 Postman으로) 응답 예시를 확인하고, 반환되는 JSON에서 필요한 데이터 추출 방법을 이해한다.	OpenAI API 간단 호출 실험 완료 (프롬프트에 대한 응답 JSON 확인)	2 시 간
Day 9	n8n에서 OpenAI 연결 및 기본 콘텐츠 생성	n8n 워크플로우에 OpenAI 노드를 추가하거나 HTTP 요청 노드로 OpenAI API를 연동한다. AppSheet에서 받은 입력 값을 활용해 프롬프트를 구성하고, OpenAI API를 호출하여 강의 계획 초안 텍스트를 생성한다. 워크플로우를 실행해 결과를 로그로 확인한다.	n8n 워크플로우에서 AppSheet 입력 → OpenAI 응답 텍스트 생성 성공 (강의 계획 초안 출력)	2 시 간
Day 10	OpenAI 프롬프트 최적화 및 출력 품질 개선	OpenAI가 반환하는 강의 계획 내용의 품질을 높이도록 프롬프트를 다듬는다. 원하는 강의 계획서 형식(예: 개요, 세부 목차 등)이 나오도록 지시문을 추가하고 여러 입력 사례로 테스트한다. 필요하다면 n8n에서 응답 텍스트를 가공하거나 불필요한 부분을 제거해 정돈한다.	개선된 프롬프트로 일관된 형식의 강의 계획 텍스트 생성 (테스트 출력물 확보)	2 시 간
Day 11	강의 계획서 결과물 Google 문서로 자동 저장	n8n에 Google Docs 노드를 연결하여 OpenAI가 생성한 강의 계획 텍스트를 문서로 자동 생성한다. Google 계정 인증을 설정하고, 워크플로우에서 새 Google Doc 파일을 만들고 제목 및 본문에 강의 계획 내용을 삽입한다. 생성된 문서의 형식을 확인하고 필요한 경우 서식을 조정한다.	OpenAI 결과물로 Google Docs 문서 생성 성공 (강의 계획서 파일 자동 생성 확인)	2 시 간
Day 12	자동 PPT 생성 구현 (Google Slides 연동)	n8n 워크플로우에 Google Slides 노드를 추가하여 강의 자료(PPT)를 자동 생성한다. Google Slides API 사용을 위해 인증을 마치고, 미리 준비한 슬라이드 템플릿 파일을 복사하거나 새 프레젠테이션을 생성한다. OpenAI로부터 얻은 강의 개요 내용을 슬라이드에 삽입 (예: 템플릿의 플레이스홀더 텍스트 치환)하고, 워크플로우 실행으로 슬라이드 파일이 만들어지는지 확인한다.	강의 주제에 대한 Google Slides 파일 자동 생성 완료 (내용이 채워진 PPT 파일 확인)	2 시 간

날짜 (Day 1~28)	학습 목표	학습/실습 내용	실습 과제 또는 산출물	예 상 소 요 시 간
Day 13	이메일 전송 자동화 (생성 자료 공유)	n8n에 Gmail 노드를 추가하여 결과물을 이메일로 전송하는 단계를 구현한다. 수신자 이메일을 설정하고, 메일 제목/본문을 작성하여 강의 계획서 요약과 Google Docs/Slides 링크를 포함한다. 필요한 경우 문서와 슬라이드를 PDF로 변환하여 첨부하고, 워크플로우 전체를 실행해 입력부터 이메일 발송까지 모두 자동화되는지 테스트한다.	자동 이메일 발송 워크플로우 완성 (강의 계획서 및 슬라이드 링크/첨부 포함된 이메일 수신 확인)	2 시 간
Day 14	2주차 리캡 및 중간 산출물 정리 (일요일)	2주 동안 구축한 OpenAI 연동 자동화 파이프라인의 흐름을 정리하고 이해도를 높인다. AppSheet 입력 → OpenAI 생성 → Docs/Slides 저장 → 이메일 전송까지 단계별로 점검하고, 개선이 필요한 부분을 목록화한다. 현재까지 완성된 기능과 산출물을 포트폴리오에 정리하여 중간 결과물과 적용 기술을 설명한다.	2주차 진행 내용 포트폴리오 업데이트 (중간 결과 요약 및 스크린샷 첨부)	2 시 간
Day 15	Google Apps Script를 활용한 슬라이드 제어 보완	Google Apps Script로 Google Slides를 프로그래밍 방식으로 다루는 방법을 학습한다. SlidesApp 서비스를 이용해 프레젠테이션을 생성하고 텍스트를 삽입하는 간단한 스크립트를 작성하여, 강의 개요 텍스트를 여러 슬라이드에 자동 배치해본다. 스크립트를 실행해 슬라이드가 원하는 형식으로 생성되는지 확인하고, 추후 n8n과의 연계를 고려한다.	Google Apps Script로 슬라이드 자동 생성 스크립트 구현 및 테스트 완료 (예: 스크립트 실행 결과 슬라이드 생성 확인)	2 시 간
Day 16	Canva AI 활용 슬라이드 디자인 탐색	Canva의 AI 기능(Magic Presentation 등)을 사용해 강의 주제만으로 자동 슬라이드 디자인을 생성해본다. Canva에 강의 주제나 요점을 입력하여 생성된 슬라이드 초안을 확인하고, 이를 n8n+Google Slides로 만든 결과와 디자인 측면에서 비교한다. Canva AI의 디자인 요소나 이미지 활용을 참고하여 향후 슬라이드 개선에 아이디어를 얻는다.	Canva AI로 생성된 슬라이드 샘플 확보 및 분석 (디자인 상의 장단점 파악 및 개선점 도출)	2 시 간
Day 17	Make 플랫폼 익히기 및 간단 자동화 시나리오	Make (Integromat)에 가입하여 기본 UI와 시나리오 작성법을 익힌다. 간단한 시나리오를 구성하여 Webhook 또는 예약 트리거로 OpenAI API를 호출하고 결과를 이메일로 보내는 흐름을 만들어본다. 이 과정에서 Make의 모듈 연결 방식과 n8n 워크플로우의 차이를 경험하고, Google Sheets나 Gmail 모듈 등도 시도해본다.	Make를 활용한 OpenAI→이메일 연동 시나리오 구현 및 실행 확인 (타 플랫폼 경험)	2 시 간

날짜 (Day 1~28)	학습 목표	학습/실습 내용	실습 과제 또는 산출물	예 상 소 요 시 간
Day 18	Make와 Canva 연계를 통한 디 자인 자동화	Make의 Canva 통합 모듈을 활용하여 Canva 디자인을 자동화하는 실험을 진행한 다. 미리 준비된 Canva 템플릿에 AppSheet/강의 정보를 채워 이미지를 생성 하고, 이를 Google Drive에 저장하거나 슬 라이드 표지로 활용하는 시나리오를 테스트 한다. Canva 연계 자동화가 강의자료 제작 워크플로우에 줄 수 있는 이점을 검토한다.	Make를 통한 Canva 디자 인 자동 생성 예제 실행 및 결과 확인 (슬라이드 시각 적 품질 향상 가능성 평가)	2 시 간
Day 19	워크플로우 개선 및 예외 처리 강 화	전체 n8n 워크플로우를 점검하며 오류 또는 예외 상황에 대한 처리를 추가한다. OpenAI 응답 실패 시 재시도 또는 알림을 설정하고, 슬라이드/문서에 예상치 못한 값이 들어갈 경우를 대비한 검증 로직을 넣는다. 또한 생 성되는 문서와 슬라이드의 품질(형식, 길이 등)을 다시 한번 다듬고 파일 이름 규칙을 적 용하여 관리하기 쉽게 개선한다.	강화된 워크플로우 버전 완성 (예외 처리 로직 적용 및 출력물 품질 개선) 및 여러 테스트 케이스 검증 완료	2 시 간
Day 20	통합 시스템 최 종 테스트 및 피 드백	AppSheet 앱 입력부터 이메일 발송까지 전 과정을 실제 시나리오로 실행하여 최종 점검 한다. 주변 동료나 지인에게 AppSheet 앱을 사용해 보도록 하고, 결과로 받은 강의 계획 서와 슬라이드를 검토받아 피드백을 수렴한 다. 모든 구성요소(계정 권한, API 호출 등)가 원활히 작동하는지 확인하고 부족한 부분을 최종 수정한다.	실제 사용 시나리오 기반 최종 테스트 완료 (피드백 반영하여 워크플로우 조정 및 버그 수정)	2 시 간
Day 21	3주차 리캡 및 포 트폴리오 업데이 트 (일요일)	3주차에 학습한 고급 도구 활용(GAS, Canva, Make)과 워크플로우 개선 내용을 정 리한다. 완성된 시스템의 구성 요소와 사용 기술을 목록화하고, 추가로 시도한 기능(디자 인 자동화 등)의 결과와 채택 여부를 기록한 다. 포트폴리오 문서를 업데이트하여 최종 프 로젝트의 기술 스택과 구현 내용을 상세히 설 명한다.	3주차 학습/개선 내용 포 트폴리오 반영 (최종 시스 템 설명 자료 업데이트)	2 시 간
Day 22	시스템 배포 및 운영 환경 구성	완성된 워크플로우를 지속적으로 실행하기 위한 운영 환경을 마련한다. n8n 워크플로우 를 클라우드 서버 등에 배포하거나 백그라운 드 실행 설정을 해서 항상 동작하도록 구성한 다. AppSheet 앱의 공유 설정과 접근 권한 을 검토하고, OpenAI 및 Google API 키/인 증 정보를 안전하게 관리하도록 환경 변수를 적용한다.	운영 환경 설정 완료 (n8n 상시 실행 설정, AppSheet 앱 배포 설정 및 API 키 보안 관리 적용)	2 시 간

날짜 (Day 1~28)	학습 목표	학습/실습 내용	실습 과제 또는 산출물	예 상 소 요 시 간
Day 23	성능 최적화 및 비용 관리	워크플로우의 성능과 API 사용량을 점검하여 최적화한다. OpenAI API 호출 빈도와 모델을 분석해 필요 시 gpt-3.5-turbo 등 비용 효율적인 모델로 조정하고, 프롬프트를 최적화하여 토큰 사용량을 줄인다. Google API 쿼타를 확인하고, 구글 서비스 호출을 최소화하도록 워크플로우 단계를 개선한다.	모델/프롬프트 최적화 및 API 사용량 조정 완료 (예상 비용 및 성능 개선 사항 정리)	2 시 간
Day 24	최종 발표 자료 준비 및 산출물 정리	프로젝트의 전 과정을 소개하는 최종 자료를 준비한다. AppSheet 앱 화면, 생성된 강의 계획서 및 슬라이드 등 핵심 산출물을 스크린샷으로 캡처하고, n8n 워크플로우와 핵심 코드(GAS 등)의 일부를 포함하여 설명용 슬라이드나 문서를 만든다. 프로젝트 개요, 사용 기술, 구현 방법, 결과를 정리한 소개 문서를 작성하여 공유 준비를 마친다.	프로젝트 소개/시연 자료 완성 (발표 슬라이드 또는 문서로 프로젝트 내용 정리)	2 시 간
Day 25	추가 개선 또는 기능 확장 (선택)	시간 여유가 있다면 프로젝트의 확장 기능을 시도한다. 예를 들어 다국어 강의 계획 지원 (한국어/영어 출력 전환)이나 슬라이드에 자동 이미지 추가 등의 기능을 하나 선정하여 구현하고 기존 워크플로우에 통합해본다. 새 기능이 정상 작동하면 결과를 확인하고, 최종 시스템을 다시 한번 전체 테스트한다.	(옵션) 추가 기능 구현 및 통합 테스트 완료 (새로운 기능 결과 확인 및 문서화)	2 시 간
Day 26	포트폴리오 최종 정리 및 공개 준 비	4주간의 학습 과정을 모두 아우르는 포트폴리오 문서를 완성한다. 프로젝트 소개, 사용한 기술 스택, 구현 과정, 어려움과 해결, 최종 산출물과 개선 내용을 상세히 작성한다. 완료된 포트폴리오를 블로그나 커뮤니티 등에 게시하거나, GitHub 레포지토리에 코드 (n8n 워크플로우 JSON, Apps Script 등)와 README를 정리하여 업로드한다.	최종 포트폴리오 작성 및 배포 완료 (프로젝트 설명 글 및 관련 코드 공개)	2 시 간
Day 27	학습 결과 회고 및 향후 계획	한 달간의 학습 성과를 돌아보고 자기 평가를 진행한다. 초기 목표 대비 달성도를 평가하고, 이번 프로젝트를 통해 성장한 점과 보완할 점을 정리한다. 향후 응용 가능성(다른 강의 주제나 분야 적용, 또는 추가 자동화 프로젝트)을 생각해보고 아이디어를 기록한다. 이번에 익힌 AppSheet, n8n, OpenAI 등 기술들의 추가 학습 계획도 세워본다.	학습 회고록 작성 및 향후 활용 계획 수립 (프로젝트 성과와 개인 성장 기록)	2 시 간

날짜 (Day 1~28)	학습 목표	학습/실습 내용	실습 과제 또는 산출물	예 상 소 요 시 간
Day 28	최종 점검 및 마무리 (일요일)	완성된 포트폴리오와 프로젝트 결과물을 최종 점검하고 제출 또는 공개한다. 모든 산출물을 백업하고, 사용한 API 키나 테스트 데이터 등 민감 정보를 정리 및 폐기한다. 4주간의 노력에 대해 자축하고, 필요하다면 프로젝트를 지인이나 온라인에 공유하며 피드백을 받아본다.	최종 결과물 제출/공유 및 프로젝트 마무리 (백업 완료, 피드백 수령)	2 시 간