



바이트디그리
iOS Development Essentials

Syllabus
Fast Campus
2020. 11

00. Byte Degree Description

바이트 디그리란, 직장인 실무 교육을 운영하는 패스트캠퍼스가 산업 전문가들과 협력을 통해 구성한 기술 인증 교육 코스입니다. 디지털 기술의 빠른 변화 속에서 현업 전문가가 설계한 프로젝트를 기반으로 학습하고, 단순히 귀와 눈으로만 익히는 것에서 벗어나 수강생 손으로 직접 구현해보며 해당 기술을 익히고, 프로젝트를 통해 이해한 정도를 검증할 수 있도록 합니다. 수강생이 학습한 내용을 바탕으로 제출하는 프로젝트는 리뷰어에게 1:1 로 피드백을 받으며 일정 수준을 통과해야 바이트 디그리를 받을 수 있습니다. 전통적인 대학 학위보다 적은 시간을 들여 실무에 가장 가까운 기술을 익히고, 프로젝트 결과물을 통해 이해도를 검증하고, 인증받을 수 있습니다.

01. Course Description

앱 개발 중에서도 제일 핫한 iOS 앱 개발의 Swift와 Xcode 학습 그리고 앱스토어 심사 과정을 경험할 수 있는 바이트디그리 Lvl. 2 과정입니다.

첫 6주에서는 총 3개의 파트로 나뉘어 학습을 하게됩니다.

- 첫번째 파트인 Swift & Xcode 학습에서는 iOS 앱 개발자는 무엇을 하는 직무인지에 대해 개념을 정리하고 iOS 앱 개발 전용 언어 Swift와 iOS 앱 개발 툴인 Xcode에 대해 학습합니다. 스위프트의 기초, Flow Control, Function and Optional, Collection, Structure 그리고 Class 와 같은 스위프트의 기본 문법과 Xcode 사용법을 알게됩니다.
- 두번째 파트인 iOS 앱 구현에서는 iOS 앱의 구조를 더 이해하고 여러 기본 앱들을 구현합니다. 해당 파트에서 수강생이 구현하게 될 앱은 현상금 추천 앱, 애플뮤직st 뮤직앱, Todo 리스트 앱입니다.
- 세번째 파트인 UI Design에서는 나중 Final Project 단계에 들어가기전 알아야 할 개발 이외의 분야인 UI 디자인에 대해 학습합니다. UI Design 파트의 목표는 Apple 심사 기준에 맞는 아이콘, 스크린, 페이지, 콘텐츠 리스트를 제작하는 법을 배우는 것입니다.

마지막 4주에서는 최종 파트인 Final Project 기간에 들어가게 됩니다.

- 프로젝트 주제 선정은 과정 개강 후, 첫번째 주에 정하게 됩니다.
- 해당 4주는 오직 Final Project를 위한 기간이며, 패스트캠퍼스가 제시하는 프로젝트 주제 4개 중 하나의 주제를 골라 본인의 창의력을 살려 실제 앱으로 구현할 수 있게 업계 최고의 개발자분들을 리뷰어로 모셔 1:1 코칭을 받게 됩니다.
- 4주안에 만들어진 앱은 실제 App Store에 등록을 하기위한 심사신청 준비를 마치게되며 App Store 심사 신청을 최소 2번 수행한 수강생들에게 본 바이트디그리의 인증서를 받게 됩니다.

본 과정은 특별 혜택으로 향후 다른 개발자와 협업할 때 쓰일 Github, Git 사용법과 iOS 기술면접에 단골로 나오는 주제를 정리한 문서를 추가 제공합니다.

02. Course Information

1. 모든 과정(강의 수강, 질의 응답, 프로젝트 피드백)은 온라인으로 진행됩니다.
2. 퀴즈 점수(출석 점수), 파트별 미니 프로젝트(2개), 최종 프로젝트 점수(1개)를 종합하여 평가 합니다.
※ UI 디자인 파트는 미니 프로젝트로 출제되지 않습니다.
3. 리뷰어를 통한 질의응답과 프로젝트 결과물에 대한 꼼꼼한 피드백을 진행합니다.

03. Course Objectives

1. Swift 언어를 이해하고 Xcode를 통해 어떻게 앱을 구현할 수 있는지에 대해 학습합니다.
2. 깊은 iOS 개발 지식없이 쉽게 구현할 수 있는 앱을 3가지 실습해봅니다.
3. 10주 동안 바이트디그리 : iOS Development Essentials를 통해 스위프트 기술을 익히고, 자신이 원하는 앱을 구현하여 앱 스토어 심사 할 수 있게 도와줍니다.

04. Course Task

1. 퀴즈 : 매주(6주) 한번씩 퀴즈를 통해 학습 한 내용을 점검합니다. 퀴즈는 객관식, 단답식, O/X 등 다양한 형태로 출제되며 학습한 내용을 복습하는 정도의 수준이 출제됩니다. 5점 만점으로 출제가 되며 4점 이상 득했을 시, 한 주 출석이 인정됩니다. 총 수업 주차(6주) 중 5주 출석 인증이 되어야 바이트 디그리 수료 조건을 충족합니다.
2. Mini Project : 총 3개 파트 중 Swift & Xcode 학습, iOS 앱 구현 2개의 파트가 끝날 때마다 각각의 파트에 해당하는 Mini Project가 출제됩니다. 해당 파트에 대한 이해도를 점검하는 수준의 프로젝트가 출제됩니다. 총 출제 되는 Mini Project 2개를 제출하여야 바이트 디그리 수료 조건을 충족합니다.
3. Final Project : 총 3개 파트 중 마지막 UI 디자인 파트의 수업이 끝나고 출제되는 프로젝트입니다. Mini Project와 동일한 형태로 문제가 출제 되지만, 모든 파트에 대한 종합적 이해가 요구되는 수준으로 Mini Project 보다 난이도 있게 출제됩니다. 제출한 프로젝트를 최종적으로 App Store 심사를 두 번 수행 하였을 시, 바이트 디그리가 수여됩니다.

05. What You Will Learn

1. Final Project를 통해 스스로 개발한 앱을 App Store 심사까지 받을 수 있게 도와줍니다.
2. iOS 앱 개발 학습을 더불어 UI 디자인에 대해서도 기본적으로 학습할 수 있습니다.
3. 수강생들은 특별 혜택으로 1) Github, Git 사용법과 2) iOS 기술면접에 단골로 나오는 주제 문서를 한번에 받을 수 있습니다.

06. Course Summary

구분	파트	파트 목표	강사	감수자
Part 1	Swift & Xcode 학습	• iOS 개발자란 무엇인가 이해하기 • Swift & Xcode 11 학습하기	이준원	김준성 전수열
Part 2	iOS 앱 구현	• 총 3가지의 앱 구현해보기: ◦ 현상금 앱, Todo 리스트, 음악 앱	이준원	김준성 전수열
Part 3	UI 디자인	• 앱 등록 신청시에 제출해야하는 아이콘, 스크린, 로그인 페이지, 콘텐츠 리스트 제작하는 법 학습하기	이영주	김준성 전수열

07. Course Schedule

- 총 수업 주차 : 10주 (수업 6주, 최종 프로젝트 4주)
- 총 수업 시간 : 1200 m

주차	기간	수업 시간(m)	특이사항
1	2020. 11. 23 ~ 2020. 11. 27	172	-
2	2020.11. 30 ~ 2020. 12. 4	197	Part 1 Mini Project
3	2020. 12. 7 ~ 2020. 12. 11	217	-
4	2020. 12. 14 ~ 2020. 12. 18	244	Part 2 Mini Project
5	2020. 12. 21 ~ 2020. 12. 25	188	-
6	2020. 12. 28 ~ 2021. 1. 1	180	-
최종 프로젝트(4주)	2021. 1. 4 ~ 2021. 1. 29	-	App Store 2번의 심사 마감 (1/29)

08. Course Project

- 총 프로젝트 : 3개 (Mini Project 2개 / Final Project 1개)
- 바이트 디그리 수료 기준 : 3개 (Mini Project 2개 제출 / Final Project 1개 통과)

1. Mini Project

a. Swift & Xcode 학습

- 출제자 : 김준성 (카카오뱅크)
- 기한 : 2020. 12. 04. 23:59
- 프로젝트 명 : 내가 좋아하는 이상형 찾기!
- 프로젝트 설명 : 이상형인지 아닌지를 알려주는 함수 만들기. 사람 객체를 만들고, 이 사람객체를 알규먼트로 받는 함수를 만들어서, 각각 attribute에 해당하는것을 점수로 변환, 이 총 점수가 20 점수보다 높으면 true, 낮으면 false로 변환하는 함수를 만듭니다.

b. iOS 앱 개발

- 출제자 : 김준성 (카카오뱅크)
- 기한 : 2020. 12. 18. 23:59
- 프로젝트 명 : 간단한 카카오톡 채팅 목록 만들기
- 프로젝트 설명 : 카카오톡 채팅 목록 화면을 만듭니다. 탭바와 상단부분을 제외하고, 채팅목록만 만듭니다. 1:다 채팅방은 제외하고, 1:1 채팅방만 있다고 가정하고 만듭니다. 이모티콘도 나타나지않고, 마지막으로 쓴 글(text)만 보인다고 가정합니다.

2. Final Project (택 1)

a. 나만의 iOS 앱 개발

- 출제자 : 김준성 (카카오뱅크)
- 기한 : 2021. 1. 29.
- 프로젝트 명 : 스파르타 가게부 (AccountBook)
- 프로젝트 설명 : 한달 예산을 설정하고, 이 한도에 얼마나 가까이 갔는지 파악해주는 어플. 돈 쓰는거에 유형을 달아서 어떤 부분에서 지출이 큰지 경고를 해줍니다.

b. 나만의 iOS 앱 개발

- 출제자 : 김준성 (카카오뱅크)
- 기한 : 2021. 1. 29.
- 프로젝트 명 : 아낌 없는 레시피 (GenerousRecipe)
- 프로젝트 설명 :남는 재료를 가지고, 만들 수 있는 음식을 찾아서 보여주는 어플입니다.

c. 나만의 iOS 앱 개발

- 출제자 : 김준성 (카카오뱅크)

- ii. 기한 : 2021. 1. 29
 - iii. 프로젝트 명 : 야너두 만들 수 있어 (AdvancedToDoApp)
 - iv. 프로젝트 설명 : Microsoft의 To Do 레벨의 투두리스트 어플을 만들어봅니다. 단순히 투두만 만들어서 처리하는게 아니라, 디테일 페이지도 추가해서 노트도 추가하고, 이미지를 추가 하는 상황도 다루게 되고, 별표처리로 우선순위를 줄 수 있습니다.
- d. 나만의 iOS 앱 개발
- i. 출제자 : 김준성 (카카오뱅크)
 - ii. 기한 : 2021. 1. 29
 - iii. 프로젝트 명 : 개인맞춤 맛집 지도 (PersonalHiddenGem)
 - iv. 프로젝트 설명 : 지도를 이용해서, 핀으로 해당하는 coordinate에 맛집을 추가합니다. 해당하는 맛집 자체에 별점을 주어서 나중에 별점을 기준으로 볼 수 있습니다.

09. Byte Degree | iOS Development Essentials 수료 기준

1. 5회 출석 체크
 - a. 총 10주 동안 6번의 출석 체크를 진행하고 5회 이상 출석 인증을 해야 합니다.
 - b. 출석 체크는 퀴즈 통과 여부를 통해 인증합니다.
 - c. 매주 수업이 끝날 때, 5점 만점의 문제가 출제 됩니다.
 - d. 4점 이상 득할 시, 1회 출석이 인정됩니다.
2. 2개 Mini Project 제출
 - a. Swift & Xcode 학습, iOS 앱 구현 2개의 파트에서 각각 1개씩 Mini Project가 출제 됩니다.
 - b. 총 2개의 Mini Project를 제출 하여야 합니다.
 - c. 제출된 Mini Project는 리뷰어를 통해 피드백을 진행합니다.
3. 1개 Final Project 통과
 - a. 모든 수업이 종료된 이후 4주의 Final Project 출제 기간이 주어집니다.
 - b. 리뷰어를 통해 피드백이 진행되고, App Store 심사를 두 번 거쳐야지 통과 기준이 됩니다.
 - c. 리뷰어의 '통과' 가 이루어진 수강생 중 출석체크, Mini Project 제출 수료 기준을 만족한 수강생에게 Byte Degree가 발급됩니다.

10. Course Curriculum

파트	상세 클립 명
Week 1: Swift & Xcode 학습	01. Ch 01. 앱 개발자 되기 - 01. iOS 앱 개발자 되기
	02. Ch 01. 앱 개발자 되기 - 02. iOS 개발 - 1
	03. Ch 01. 앱 개발자 되기 - 02. iOS 개발 - 2
	04. Ch 01. 앱 개발자 되기 - 02. iOS 개발 - 3
	01. Ch 03. 스위프트 기초 - 01. 스위프트 기초
	02. Ch 03. 스위프트 기초 - 02. 플레이그라운드
	03. Ch 03. 스위프트 기초 - 03. 코멘트
	04. Ch 03. 스위프트 기초 - 04. 튜플
	05. Ch 03. 스위프트 기초 - 05. Boolean
	06. Ch 03. 스위프트 기초 - 06. Boolean 추가개념
	07. Ch 03. 스위프트 기초 - 07. Scope
	01. Ch 04 스위프트 Flow Control - 01. 플로우컨트롤 인소개
	02. Ch 04. 스위프트 Flow Control - 02. while
	03. Ch 04. 스위프트 Flow Control - 03. for loop 기본
	04. Ch 04. 스위프트 Flow Control - 04. for loop 심화
	05. Ch 04. 스위프트 Flow Control - 05. switch 기본
	06. Ch 04. 스위프트 Flow Control - 06. switch 심화
Week 2: Swift & Xcode 학습	01. Ch 05. 스위프트 Function and Optional - 01. 함수소개
	02. Ch 05. 스위프트 Function and Optional - 02. 함수 실습 -1
	03. Ch 05. 스위프트 Function and Optional - 03. 함수 실습 -2
	04. Ch 05. 스위프트 Function and Optional - 04. 함수 요약
	05. Ch 05. 스위프트 Function and Optional - 05. 도전과제
	06. Ch 05. 스위프트 Function and Optional - 06. 고급기능 -1
	07. Ch 05. 스위프트 Function and Optional - 07. 고급기능 -2

	08. Ch 05. 스위프트 Function and Optional - 08. 옵셔널 기초
	09. Ch 05. 스위프트 Function and Optional - 09. 옵셔널 기초 실습
	10. Ch 05. 스위프트 Function and Optional - 10. 옵셔널 고급 -1
	11. Ch 05. 스위프트 Function and Optional - 11. 옵셔널 고급 -2
	12. Ch 05. 스위프트 Function and Optional - 12. 옵셔널 도전과제
	01. Ch 06. 스위프트 Collection - 01. 컬렉션 소개
	02. Ch 06. 스위프트 Collection - 02. Array 개념
	03. Ch 06. 스위프트 Collection - 03. Array 실습 -1
	04. Ch 06. 스위프트 Collection - 04. Array 실습 -2
	05. Ch 06. 스위프트 Collection - 05. Dictionary 개념
	06. Ch 06. 스위프트 Collection - 06. Dictionary 실습
	07. Ch 06. 스위프트 Collection - 07. Set 개념과 실습
	08. Ch 06. 스위프트 Collection - 08. Closure 개념과 실습
	09. Ch 06. 스위프트 Collection - 09. Closure - Capturing Value
Week 3: Swift & Xcode 학습	01. Ch 07. 스위프트 Structure - 01. 구조체와 클래스 차이
	02. Ch 07. 스위프트 Structure - 02. 구조체 실습
	03. Ch 07. 스위프트 Structure - 03. 구조체 도전과제
	04. Ch 07. 스위프트 Structure - 04. 프로토콜
	05. Ch 07. 스위프트 Structure - 05. 프로퍼티 개념
	06. Ch 07. 스위프트 Structure - 06. 프로퍼티 실습 - 1
	07. Ch 07. 스위프트 Structure - 07. 프로퍼티 실습 - 2
	08. Ch 07. 스위프트 Structure - 08. 프로퍼티 vs 메소드
	09. Ch 07. 스위프트 Structure - 09. 메소드 개념 및 실습
	10. Ch 07. 스위프트 Structure - 10. 메소드 확장
	01. Ch 08. 스위프트 Class - 01. 클래스 개념
	02. Ch 08. 스위프트 Class - 02. 클래스 실습
	03. Ch 08. 스위프트 Class - 03. 언제 클래스 혹은 스트럭트를 써야 할까
	04. Ch 08. 스위프트 Class - 04. 상속개념 코드로 바로 배우기
	05. Ch 08. 스위프트 Class - 05. 상속의 규칙과 실습
	06. Ch 08. 스위프트 Class - 06. 클래스 상속은 언제 쓸까
	07. Ch 08. 스위프트 Class - 06. 생성자 이해하기 #1
	08. Ch 08. 스위프트 Class - 06. 생성자 이해하기 #2
	09. Ch 08. 스위프트 Class - 06. 생성자 이해하기 #3
Week 4: iOS 앱 구현	01. Ch 09. 원피스 현상금 랭킹앱 - 01. 아이폰개발자가 알면 좋은 경제지식
	02. Ch 09. 원피스 현상금 랭킹앱 - 02. 테이블뷰 기초 개념
	03. Ch 09. 원피스 현상금 랭킹앱 - 03. 테이블뷰 프로토콜 개념
	04. Ch 09. 원피스 현상금 랭킹앱 - 04. 테이블뷰 프로토콜 실습
	05. Ch 09. 원피스 현상금 랭킹앱 - 05. 커스텀 테이블뷰 셀 만들기
	06. Ch 09. 원피스 현상금 랭킹앱 - 06. 뷰컨트롤러 모달로 띄우기
	07. Ch 09. 원피스 현상금 랭킹앱 - 07. 뷰컨트롤러간 데이터 전달하기
	08. Ch 09. 원피스 현상금 랭킹앱 - 08. 코드 회고

	01. Ch 13. Todo 리스트 앱 - 01. todoList 앱 소개
	02. Ch 13. Todo 리스트 앱 - 02. 탭바 컨트롤러 구성하기
	03. Ch 13. Todo 리스트 앱 - 03. 스택틱 테이블뷰로 설정페이지 구현하기
	04. Ch 13. Todo 리스트 앱 - 04. 태스크 관리뷰, 컬렉션뷰 구현하기 01
	05. Ch 13. Todo 리스트 앱 - 05. 태스크 관리뷰, todolist셀 기능 확인 및 투두 객체 소개
	06. Ch 13. Todo 리스트 앱 - 06. 태스크 관리뷰, todoManager 소개 및 구현하기
	07. Ch 13. Todo 리스트 앱 - 07. 태스크 관리뷰, todolist뷰모델 이해하고 구현하기
	08. Ch 13. Todo 리스트 앱 - 08. 태스크 관리뷰, 컬렉션뷰 구현하기 02
	09. Ch 13. Todo 리스트 앱 - 09. 태스크 관리뷰, struct 를 json으로 인코딩 시켜서 disk에 저장해보기
	10. Ch 13. Todo 리스트 앱 - 10. 태스크 관리뷰, 텍스트 인풋 뷰 추가
	11. Ch 13. Todo 리스트 앱 - 11. 태스크 관리뷰, 키보드에 따라 인풋뷰 위치 변경
	12. Ch 13. Todo 리스트 앱 - 12. 태스크 관리뷰, 투두 추가하기, 체크버튼 구현, 삭제 버튼 구현, 다크모드 확인
Week 5: iOS 앱 구현	01. Ch 12. 애플뮤직st 음악앱 - 01. 뮤직앱 소개
	02. Ch 12. 애플뮤직st 음악앱 - 03. 홈화면 컬렉션뷰 추가
	03. Ch 12. 애플뮤직st 음악앱 - 02. 뮤직앱 프로젝트 구조 확인
	04. Ch 12. 애플뮤직st 음악앱 - 04. 홈화면 트랙매니저 구현 01
	05. Ch 12. 애플뮤직st 음악앱 - 05. 홈화면 트랙매니저 구현 02
	06. Ch 12. 애플뮤직st 음악앱 - 06. AVPlayerItem을 확장시켜보기(extension)
	07. Ch 12. 애플뮤직st 음악앱 - 07. 홈화면 컬렉션뷰 셀 구현 마무리
	08. Ch 12. 애플뮤직st 음악앱 - 08. 홈화면 컬렉션뷰 헤더 구현 01
	09. Ch 12. 애플뮤직st 음악앱 - 09. 홈화면 컬렉션뷰 헤더 구현 02
	10. Ch 12. 애플뮤직st 음악앱 - 10. 플레이어뷰 뷰구성하기
	11. Ch 12. 애플뮤직st 음악앱 - 11. 플레이어뷰 플레이어 만들기
	12. Ch 12. 애플뮤직st 음악앱 - 12. 홈화면에서 플레이어화면 띄우기
	13. Ch 12. 애플뮤직st 음악앱 - 13. 플레이어뷰 곡정보 표시 및 재생구현하기
	14. Ch 12. 애플뮤직st 음악앱 - 14. 플레이어뷰 재생시간 실시간 업데이트 및 시킹 구현
	15. Ch 12. 애플뮤직st 음악앱 - 15. 플레이어뷰 다크모드 설명 및 앱전체 요약
Week 6: UI Design	01. Ch 01. 강의를 시작하며- 01. 강의 소개, 방향 안내
	02. Ch 01. Human Interface Guidelines - 02. Themes 테마
	03. Ch 01. Human Interface Guidelines - 03. Interface Essentials 인터페이스 필수요소
	04. Ch 01. Human Interface Guidelines - 04. App Architecture 앱의 구조
	05. Ch 02. Figma랑 친해지기 - 05. Figma를 선택한 이유
	06. Ch 02. Figma랑 친해지기 - 06. 기본 인터페이스, 디자인 공유 방식
	07. Ch 02. Figma랑 친해지기 - 07. Component
	08. Ch 02. Figma랑 친해지기 - 08. Auto Layout
	09. Ch 02. Figma랑 친해지기 - 09. Plugin
	10. Ch 03. 디자인 시작하기 - 10. 참고 자료 찾기
	11. Ch 03. 디자인 시작하기 - 11. 디자인 방향을 정하기
	12. Ch 03. 디자인 시작하기 - 12. 무드보드 만들기
	13. Ch 04. 로그인 페이지 디자인 - 13. 레이아웃 잡기

	14. Ch 04. 로그인 페이지 디자인 - 14. Text Fields 디자인 하기
	15. Ch 04. 로그인 페이지 디자인 - 15. Button 디자인 하기
	16. Ch 05. 콘텐츠 리스트 디자인 - 16. 리스트 UI의 특징, 디자인에 일관성을 생기게 하려면?
	17. Ch 05. 콘텐츠 리스트 디자인 - 17. 레이아웃 잡기
	18. Ch 05. 콘텐츠 리스트 디자인 - 18. 컴포넌트 & 오토레이아웃으로 디자인 하기
	19. Ch 06. 앱 아이콘 디자인 - 19. 앱 아이콘 아이디어션
	20. Ch 06. 앱 아이콘 디자인 - 20. 앱 아이콘 디자인
	21. Ch 07. 스플래시 디자인 - 21. 앱 아이콘과 연결성을 주는 스플래시
	22. Ch 07. 스플래시 디자인 - 22. 스플래시와 런치 스크린의 차이, 진입형 UI 종류
	23. Ch 07. 프로토타이핑 - 23. 간단한 프로토타입 만들기

(끝)