

Microsoft Professional Program for Entry Level Software Development

13
REQUIRED COURSES

8 - 16
HOURS PER COURSE

11
SKILLS

거의 모든 기술은 소프트웨어에 의존합니다. 소프트웨어 개발자들의 직업 전망은 해마다 증가하고 있습니다. 간단히 말해, **Software Developers** 는 현대 사회에 생명을 불어 넣는 사람들입니다.

교육 과정 소개

이 커리큘럼은 프로그래밍을 시도한 적이 없는 사람들을 위해 개발되었습니다.

기본 프로그램 작성, 대화 형 웹 페이지 개발, 사용자를 위한 디자인 구성 방법 이해, 효율성을 위한 알고리즘 및 데이터 분석을 통해 논리적이고 체계적으로 사고하는 방법을 배우게 됩니다. SW 중심 사회, SW 코딩 교육으로 인해 많은 부담감을 갖고 있는 중고등학교 학생들과 교사 선생님, 또한 대학에서 IT 비 전공자이지만, IT를 추가로 학습하여 본인의 전공 영역을 보다 확대하고자 하는 학생들에게 좋은 과정입니다.

Think Logically and Systematically (논리적이고 체계적인 사고력)



(과목 명 - **Logic and Computational Thinking** : Microsoft 제공)

이 과정은 공식 논리의 기초를 가르치고 논리적 사고와 논리적 개념이 디지털 기계 설계 및 구축에 어떻게 적용되었는지를 보여줍니다. 논리적 분석을 통해 기술을 습득하면 장기적으로 더 나은 엔지니어가 될 수 있도록 컴퓨터 알고리즘의 기초를 더 잘 이해할 수 있습니다.

Create Basic Programs and Functions (기본 프로그램 및 Function 만들기)



(과목 명 - **Introduction to Python: Absolute Beginner** : Microsoft 제공)

새로운 브랜드의 텍스트 기반 프로그래밍. 이 실습 과정을 통해 파이썬 레이어와 개념에 대한 자세한 정보를 배울 수 있습니다. Azure 에서 브라우저와 인터넷 연결만 필요한 Jupyter Notebooks 를 사용하여 실습과 프로젝트를 충분히 진행할 수 있습니다. 좋은 구현 사례를 배우고 즉시 코딩을 시작하세요.

Process Simple Data Structures and Files (간단한 데이터 구조 및 파일 처리)



(과목 명 - **Introduction to Python: Fundamentals** : Microsoft 제공)

파이썬 기술을 조금 더 깊이 배웁니다. 파이썬 데이터 구조에 중점을 두고 문자열, 리스트 및 범위 시퀀스에 대해 다룹니다. 리스트 반복의 힘을 발견하고 문자열 및 리스트 방법에 대해 배웁니다. 파일 입력 및 출력에 대한 세부 정보를 가져 와서 파일을 열고, 파일을 읽거나 추가하거나 닫는 작업 등을 배웁니다. 학습이 끝나면 코딩을 시작하면서 문자열을 하위 문자열로 나누고, 리스트를 만들고, 반복하고, 파일을 가져오고, 파일 추가 모드를 사용할 수 있을 것입니다..

Develop a Simple Technical Solution (간단한 기술 솔루션 개발)



(과목 명 - **Designing a Technical Solution** : Microsoft 제공)

아이디어를 기술적인 솔루션으로 바꾸는 과정을 배우십시오. 학습자는 고급 세부 사항에 뛰어 들기 전에 상황에 대한 지식과 프로세스상의 좋은 아이디어를 습득해야 합니다. 바로 이 과정에서 얻게 될 것입니다. 시장 조사, 사용자 중심 디자인 및 프론트 엔드 웹 개발 프로세스를 배우면서 앱에 대한 자신 만의 아이디어를 구축 할 수 있을 것입니다. 과정이 끝나면 HTML 및 CSS 로 제작 된 프론트 엔드 사용자 인터페이스와 함께 사용자 중심의 디자인, 와이어 프레임 및 웹 앱 사양이 제공됩니다.

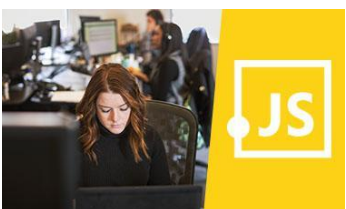
Develop Interactive Webpages (대화형 웹 페이지 개발)



(과목 명 - **Building Interactive Prototypes using JavaScript** : Microsoft 제공)

정적인 프론트 엔드를 사용자가 상호 작용할 수 있는 대상으로 전환하는 방법에 대해 알아보십시오. 자바 스크립트의 기초를 배우고 자바 스크립트를 적용하여 웹 페이지의 상호 작용을 향상시키는 방법을 배웁니다. 이 과정이 끝나면 기본적인 상호 작용이 가능한 웹 앱 프로토타입이 생깁니다.

Develop and Deploy Dynamic Websites (동적인 웹사이트 개발 및 배포)



(과목 명 - **Building Functional Prototypes using Node.js** : Microsoft 제공)

웹 개발 및 응용 프로그램 동작을 결합하여 이 과정에서는 백 엔드 서버의 개념, 웹에서의 데이터 전송 및 저장 개념, Azure 에서 웹 응용 프로그램 배포, Node.JS 를 사용하여 백 엔드 웹 서버 만들기, 데이터베이스의 용도와 유형을 배우게 됩니다.

Create Advanced Programs and Functions (고급 프로그램 및 기능 만들기)



(과목 명 - **Learn to Program in Java** : Microsoft 제공)

Java 로 코딩하는 것뿐만 아니라 가장 기본적인 컴퓨터 사이언스 도구를 사용하여 문제를 해결하고 솔루션을 구현하는 방법을 배우십시오. 과정 전반에 걸쳐 methods, logic, loops, variables, parameters, returns, recursion 에 대한 코딩 경험을 충분히 습득하게 됩니다. 업계 표준 툴과 연습을 통해 노력한다면 개발자가 갖추어야 할 스킬 셋을 성장시키고 강한 습관을 구축 할 수 있습니다.

Create Basic Object-Oriented Programs (기본 객체 지향 프로그램 만들기)



(과목 명 - **Object Oriented Programming in Java** : Microsoft 제공)

Java 를 사용하여 코드에서 객체의 역할, 최적의 코드 구조를 위해 객체를 설계하는 방법을 배웁니다. 배열 및 목록과 같이 널리 사용되는 Java 객체를 사용하여 코딩 경험을 쌓으면 자신의 객체 계층 구조를 디자인 할 수 있습니다. 상속, 인터페이스, 추상 클래스 및 캡슐화 기술을 사용하여 연습 할 수 있습니다.

Analyze Algorithms and Data Structures for Efficiency (효율적인 알고리즘 및 데이터 구조 분석)



(과목 명 - **Algorithms and Data Structures** : Microsoft 제공)

이 과정에서는 더 나은 프로그램을 만드는 법을 배웁니다. 프로그래밍 경험을 적용하고 즉시 사용 가능한 기본 개념을 자세히 살펴보십시오. 기능적 알고리즘과 데이터 구조의 원리를 깊이 있게 살펴보고 어떻게 작동하는지, 어떻게 비교하는지 확인하십시오. 또한 언제, 어떻게 사용하는지 자세히 배울 수 있습니다.

Apply Software Development to Real-World Scenarios (실제 시나리오에 소프트웨어 개발 적용)



(과목 명 - **Writing Professional Code** : Microsoft 제공)

실제 문제에 코딩 기술을 적용하는 것이 얼마나 재미 있는지 알아보십시오. GitHub 에 온라인 코드 포트폴리오를 만들고, 코드 검토 및 토론, 오픈 소스 (OSS) 라이브러리 사용, 기존 코드 업데이트, 가독성 개선 및 보안 검토를 통해 동료들의 의견을 수렴해보세요.

Design User-Centric Software (사용자 중심의 소프트웨어 설계)



(과목 명 - **Introduction to Design Thinking** : Microsoft 제공)

이 과정은 Design Thinking 과정을 소개하고 그 과정에서 각 단계의 모범 사례를 보여줍니다. 엔지니어링에서부터 기업가 정신에 이르기까지 모든 분야에서 매우 강력한 도구라는 것을 알게 될 것입니다. 당신은 이 과정에서 배운 모든 것을 활용하여 자신만의 프로젝트를 만듭니다. 그렇게 함으로써 사용자 연구와 신속한 프로토타이핑과 같은 많은 실용적이고 적용 가능한 기술을 배우게 될 것입니다.

Design Software for Global Audience (전 세계 고객을 위한 디자인 소프트웨어)



(과목 명 - **Introduction to Developing International Software** : Microsoft 제공)

처음부터 여러 언어에 대한 학습 계획을 세우는 것이 훨씬 더 효율적이며, 적은 비용으로 어떻게 배울 수 있는지 알아보십시오. 운영 체제 및 프로그래밍 언어의 국제적 기능을 활용하면 여러 언어 및 시장에서 사용할 수 있는 코드를 작성할 수 있습니다. 이 과정에는 세 가지 부분, 즉 비즈니스 사례, world-ready design, and world-ready development 가 포함됩니다. 학생들은 프로그래밍 지식없이 파트 1 과 파트 2 를 완성 할 수 있지만 파트 3 에서는 프로그래밍 경험이 필요합니다.