

# 강의계획서

## ■ 강의과목

| 교과목명 | 파이썬 기초실습 익히기                            |      |     |
|------|-----------------------------------------|------|-----|
| 강의형태 | 원격(온라인)                                 | 강의유형 | 온라인 |
| 개발역량 | 자기관리 10, 정보화 30, 종합적 사고 30, 창의적 문제해결 30 |      |     |

## ■ 교과목 개요

- 프로그래밍을 처음 접하는 입문자를 대상으로 하며, 복잡한 문제를 논리적으로 분해하고 해결하는 컴퓨팅 사고력의 원리를 파이썬 언어를 통해 습득할 수 있도록 핵심 이론과 실습 위주로 콘텐츠를 구성함.
- 변수, 연산자, 자료형 등 기초 문법부터 조건문(if)과 반복문(while, for) 등 프로그램 제어 구조를 단계별 실습을 통해 학습하며, 학습자가 스스로 논리적인 코드를 설계하고 구현할 수 있는 실무 능력을 배양하고자 함.
- 단순한 문법 암기에서 벗어나 일상 속의 다양한 문제를 소프트웨어적 관점에서 분석하고 해결하는 경험을 제공함으로써, 디지털 전환 시대에 필수적인 기초 SW 문해력을 높이고 프로그래밍에 대한 자신감을 가질 수 있도록 함.

## ■ 교육 목표

- 본 교과는 컴퓨팅 사고력을 배양하고, 컴퓨터 프로그래밍 기술을 일상적인 일반 문제에 직접 적용하여 해결하는 능력을 높이는 것을 목표로 한다. 파이썬 프로그래밍의 기초 문법부터 실전 활용까지 실습 중심으로 학습함으로써 교육생들이 논리적 사고를 코드로 구현하는 역량을 체계적으로 함양한다.
- 이 강의를 통해 학생들은 다음과 같은 능력을 갖추게 된다.
  - ① 컴퓨팅 사고력의 이해와 적용: 문제 해결을 위한 논리적 절차인 컴퓨팅 사고의 핵심 개념을 이해하고, 이를 바탕으로 프로그래밍의 기본 원리를 설명할 수 있다.
  - ② 파이썬 기초 문법 및 제어 구조 학습: 변수, 연산자, 자료형의 개념을 확립하고, 선택문(if)과 반복문(while, for) 등 다양한 제어 구조를 활용하여 역동적인 프로그램을 작성할 수 있다.
  - ③ 실전 문제 해결 및 프로그램 구현: 이론으로 배운 파이썬 기술을 실제 문제에 적용하여 스스로 코드를 설계하고 구현함으로써 디지털 시대에 필요한 실무적인 소프트웨어 활용 능력을 갖춘다.

## ■ 교육 구성

| 주차 | 주제명                          | 세부내용                                                                                                                       |
|----|------------------------------|----------------------------------------------------------------------------------------------------------------------------|
| 1  | 컴퓨터 프로그램의 이해                 | <ul style="list-style-type: none"> <li>• 컴퓨팅적 사고의 정의</li> <li>• 컴퓨적적 사고의 개념</li> <li>• 컴퓨팅적 사고 사례</li> </ul>               |
| 2  | 파이썬 시작하기                     | <ul style="list-style-type: none"> <li>• 파이썬 개요</li> <li>• 파이썬 설치하기</li> <li>• 셸, IDLE 편집기</li> </ul>                      |
| 3  | 변수와 자료형 1                    | <ul style="list-style-type: none"> <li>• 변수의 개념 및 생성 규칙</li> <li>• 변수의 타입</li> <li>• 기본 자료형(숫자형, 문자형, 불리언)</li> </ul>      |
| 4  | 변수와 자료형 2                    | <ul style="list-style-type: none"> <li>• 컬렉션 자료형의 개념</li> <li>• 리스트, 튜플, 집합, 딕셔너리</li> </ul>                               |
| 5  | 연산자와<br>표준 입출력               | <ul style="list-style-type: none"> <li>• 연산자(산술, 비교, 논리, 문자열)</li> <li>• 표준 입출력(print)</li> <li>• 표준 입출력(input)</li> </ul> |
| 6  | 단일 선택문(if) 활용                | <ul style="list-style-type: none"> <li>• 선택 구조의 개념 및 필요성</li> <li>• if문의 의미 및 사용법</li> </ul>                               |
| 7  | 단일 선택문(if, elif, else)<br>활용 | <ul style="list-style-type: none"> <li>• if~else 문의 개념 및 사용법</li> <li>• if~elif~else 문의 개념 및 사용법</li> </ul>                |
| 8  | 반복문(while) 활용                | <ul style="list-style-type: none"> <li>• 반복 구조의 필요성</li> <li>• while문의 의미 및 사용법</li> </ul>                                 |
| 9  | 반복문(for) 활용                  | <ul style="list-style-type: none"> <li>• for 문의 의미 및 사용법</li> <li>• 기타 제어문의 종류 및 활용법</li> </ul>                            |