

## 목차

### 1. 자연어와 프로그래밍 언어

#### 1-1. 자연어

#### 1-2. 프로그래밍 언어

#### 1-3. 추상화

### 2. 객체 지향적 사고

### 3. 함수적 사고

### 1. 자연어와 프로그래밍 언어

#### 1-1. 자연어

자연어는 생각을 전달하고 소통하기 위한 언어입니다. 우리의 생각은 머릿속에서 감각적인 경험을 통해 형성되며, 규칙이 있는 경우에 언어로 표현됩니다. 자연어는 세상에 대한 표현이며, 사람은 추상화를 통해 생각을 기억(저장)하고, 이해한다.

#### 1-2. 프로그래밍 언어

프로그래밍은 자연어로 받은 요구사항을 프로그래밍 언어로 변환하는 과정이다. 자연어는 모호성을 가지지만, 프로그래밍 언어는 확정적인 의미를 가져야 한다. 그래서 자연어를 모호성이 없는 프로그래밍 언어로 바꿔서 프로그램을 만들어야 한다. 특정 분야에는 그 분야를 가장 잘 표현할 수 있는 언어가 있다. 자연어는 거의 모든 표현을 담고 있지만, 프로그래밍 언어는 컴퓨터가 실행할 수 있는 특성을 갖고 있다.

객체지향언어는 자바처럼 클래스와 오브젝트를 중심으로 프로그래밍하는 방식을 말한다. 자바에서의 명사는 클래스와 오브젝트(객체), 동사는 메소드, 함수를 말한다.

자연어와 프로그래밍 언어는 세상을 표현하기 위한 도구이다. 자연어는 모호성이 있고 추상적인 면이 있는 반면에, 프로그래밍 언어는 구조적이고 확정적인 의미를 가져야 한다.

#### 1-3. 추상화

추상화는 세상에서 공통성이나 특정한 부분을 뽑아내어 타입을 만들고, 타입 안에 값을 넣어서 인스턴스를 만드는 과정이다.

세상은 다양한 관점으로 나누어 추상화되고, 우리는 특정 부분을 분석하여 배운다. 학문 분야는 이 추상화된 지식을 세계에 일반화하여 정리한 것이다. 세상은 명사와 동사로 이루어져 있다. 명사는 존재하는 것을 나타내고, 동사는 명사를 변화시키는 역할을 한다. 이러한 세상의 구성요소를 프로그래밍 언어를 사용하여 생각을 표현하고 프로그램으로 옮길 수 있다.

## 2. 객체 지향적 사고

객체 지향적 사고는 데이터를 중심으로 생각하는 방식이다. 이는 데이터와 그 데이터를 변화시키는 메소드 또는 함수를 묶어서 객체라는 단위로 표현하는 것을 말한다. “객체 지향적 사고방식을 기반으로 하는 소프트웨어 개발 기법의 하나이다. 시스템이나 시스템의 구성 요소를 메시지의 교환에 의해 상호 통신하는 객체의 집합으로 생각한다. 객체의 종류, 객체의 절차(방법), 객체 간에 절차의 수행을 위해 주고받는 메시지를 정의하는 데 중점을 두고 설계한다.” [네이버 지식백과] 객체 지향 설계 [object-oriented design, 客體指向設計] (IT용어사전, 한국정보통신기술협회)

예를 들어,

```
Integer y = new Integer();  
y.add(x);
```

```
Integer add(integer x);
```

이 add 메소드는 입력으로 정수 x를 받아들이고, 현재의 Integer 객체에 x를 더한 결과를 반환한다. 사용자는 새로운 Integer 객체를 생성하고, 그 객체의 add 메소드를 호출해서 작업을 수행할 수 있다. 이 경우에는 출력 데이터인 y가 중심이 된다.

## 3. 함수적 사고

함수적 사고는 동사를 중심으로 데이터를 변형하는 방식을 말한다. 함수는 어떤 데이터를 입력받아서 그 데이터를 변형하여 결과를 반환한다. 예를 들어,

$$y = x + z;$$

라는 식에서는 ‘+’가 동사의 역할을 하고 있다. x와 z는 입력 데이터이고, 더하기 연산을 통해 두 값을 더하여 결과를 y에 할당한다. 즉 함수적 사고 방식은 데이터를 변형하거나 조작하여 원하는 결과를 얻는 데에 중심을 두는 사고 방식이다.