1. **객체 지향 언어**
   1. 객체 지향 언어란?

객체 지향 언어는 객체 지향 프로그래밍이 가능한 언어이다. 객체지향 프로그램이란 컴퓨터 프로그램을 명령어 목록으로 보는 것에서 벗어나, 객체들의 모임으로서 파악하는 것이다. 따라서 프로그램은 각 객체들 간의 상화작용을 정의 하는 것으로 이루어진다.

* 1. 장점
* 프로그램을 유연하고 변경이 쉽게 만든다.
* 대형 프로젝트에 적합하다.
* 직관적인 코드 분석을 가능하게 한다.
  1. 단점
* 일반적으로 절차 지향보다 실행 시 속도가 느리다.
* 설계에 많은 시간이 들어간다.
  1. 절차 지향 언어와의 차이점.

|  |  |
| --- | --- |
| 절차 지향 언어 | 객체 지향 언어 |
| 함수 호출의 개념을 바탕으로 한다. | 객체 간의 상호작용을 바탕으로 한다. |
| 데이터를 중심으로 함수 구현 | 기능을 중심으로 메서드 구현 |

1. **용어 조사**
   1. 클래스

객체 지향 프로그래밍에서 객체를 생성하기 위해 변수와 메소드를 정의하는 곳이다. 객체를 생성하기 위한 설계도로 볼 수 있다.

* 1. 객체

클래스에서 정의한 것을 토대로 실제로 메모리에 할당한 것으로 생성자를 호출해 생성할 수 있다. 클래스가 설계도라면 객체는 설계도를 기반으로 만든 물건으로 볼 수 있다.

* 1. 캡슐화

캡슐화는 연관 있는 변수와 함수를 클래스로 묶어 실제 구현 내용을 외부에 감추는 것을 뜻한다. 외부 객체는 내부 구조를 얻지 못하기 때문에 객체가 노출해서 제공하는 필드와 메소드만 사용할 수 있다. 캡슐화 된 멤버에 대한 접근은 접근 제한자에 따라 결정되는데 보통 아래의 세가지가 사용된다.

* Public : 클래스 내외부에서 모두 사용가능
* Protected : 자식 클래스에서 사용가능
* Private : 클래스 내부에서만 사용 가능
  1. 상속성

상속은 새로운 클래스가 기존의 클래스의 자료와 연산을 이용할 수 있 게 하는 기능이다. 상속을 받는 새로운 클래스를 자식클래스라고 하고, 상속하는 클래스를 부모 클래스라고 한다. 자식 클래스에서는 부모에 받은 자료와 연산을 접근제한에 따라 수정할 수 있다.

* 1. 다형성

다형성은 어떤 한 요소에 여러 개념을 넣는 것으로 일반적으로 오버라이딩과 오버로딩을 의미한다.

* 오버라이딩 : 같은 이름의 메서드가 여러 클래스에서 다른 기능을 수행하는 것.
* 오버로딩 : 같은 이름의 메서드가 파라미터의 개수나 종류에 따라 다른 기능을 수행하는 것

1. **프로젝트 계획**
   1. 프로그램 목표

실제로 연구실에서 사용가능한 웹 페이지를 제작하고 활용하는 것을 목적으로 한다.

* 1. 기능
     1. 연구실 소개 페이지

간단하게 연구실을 소개하는 페이지로 현재 있는 연구원들도 함께 표시한다.

* + 1. 회원가입 기능

연구실 인원만 사용할 수 있도록 회원가입 기능을 구현한다. 회원 가입 요청 후 최종적으로 관리자가 승인해야 회원가입이 완료된다.

* + 1. 게시판

유저는 게시판에 글을 남기거나 이미 있는 글에 댓글을 남길 수 있다. 관리자는 게시판에 공지사항을 올리거나 내릴 수 있다. 또한 게시판 검색을 통해 제목, 내용, 댓글 내용을 통해 검색이 가능하다.

* + 1. 스케줄 관리 기능

유저들은 수업 같은 개인 스케줄 입력해서 관리하거나, 회의 같은 단체 스케줄을 관리할 수 있다. 개인 스케줄의 경우 다른 사람이 내용을 볼 수 있게 조절하는 ‘내용 공개’ 여부를 선택할 수 있고, 단체 스케줄은 같이 진행할 유저를 선택하여 생성할 수 있다. 일정 조정을 쉽게 하기 위해서 유저들을 선택하면 모두가 가능한 일정을 표시해준다.