|  |  |
| --- | --- |
| 교육 제목 | **Python 기초 함수 함수고급 예외처리** |
| 교육 일시 | 2021.09.24 |
| 교육 장소 | YGL-C6 |
| **교육 내용** | |
| 오전 | 1. 재귀 함수    1. 팩토리얼(!) 구하기   n! = n \* (n-1) \* (n-2) \* . . . \*1   * 1. 피보나치 수열 구하기   피보나치 수를 구할 때 재귀 함수를 사용하면 필요하지 않은 연산의 양이 많아짐  이를 해결 하기 위해 메모화 사용  텍스트이(가) 표시된 사진  자동 생성된 설명   * 1. 평탄화 함수 -> 다차원의 리스트를 1차원의 리스트로 만들어주는 함수   재귀 함수를 이용해서 구현  텍스트이(가) 표시된 사진  자동 생성된 설명   1. filter(() 함수와 map()함수   map(함수, 리스트) -> 리스트의 요소를 함수에 넣어서 수행후 리스트 객체로 반환  filter(함수, 리스트) ->리스트 값이 함수 조건에 맞는 데이터만 반환 -> 결과를 리스트 객체로 반환  텍스트이(가) 표시된 사진  자동 생성된 설명   1. lambda 함수   짧은 함수를 간단하게 구현 lambda 매개변수 : 리턴값  텍스트이(가) 표시된 사진  자동 생성된 설명   1. 파일 처리   텍스트 파일 처리  파일 열기(open()함수 이용 -> read() or write() 함수를 이용해 읽고 쓰기  텍스트이(가) 표시된 사진  자동 생성된 설명   1. with 구문   file 입출력시 close를 굳이 쓰지 않아도 스스로 file을 닫는다.  파일 입출력시 close를 하지 않는 실수를 미연에 방지 할 수 있다.  텍스트이(가) 표시된 사진  자동 생성된 설명   1. 제너레이터 함수 만들기   yield 키워드를 넣으면 일반 함수가 제너레이터 함수가 된다.  제너레이터 사용시 메모리를 효율적으로 활용 할 수 있다.  텍스트이(가) 표시된 사진  자동 생성된 설명 |
| 오후 | 1. 예외처리   핵심 키워드  구문 오류, 예외(런타임 에러), 기본 예외 처리, try except구문  구문 오류 : 실행 전 발생하는 오류  런타임 에러 : 실행중 발생하는 오류  핵심 포인트  프로그램 활용 시 쉽게 예상치 못하는 상황들에 대해 알아보고 세분화   1. 조건문으로 예외 처리   텍스트, 영수증이(가) 표시된 사진  자동 생성된 설명   1. try except문으로 예외 처리   텍스트이(가) 표시된 사진  자동 생성된 설명  **예외 처리 사용방법**   * try + except * try + except + else * try + except + finally * try + except + else + finally * try + finally   텍스트이(가) 표시된 사진  자동 생성된 설명 텍스트이(가) 표시된 사진  자동 생성된 설명  텍스트이(가) 표시된 사진  자동 생성된 설명   1. raise 구문   -프로그램이 강제 종료되는것을 막기 위해 예외는 꼭 처리 해야한다.  -프로그램 개발동안 구현되지 않는 부분일 경우 NotImplementedError를 발생 시킨다  -아직 구현되지 않는 부분을 일부로 예외를 발생 시켜 프로그램을 죽게 만들어 추후 개발에  문제가 되지 않게 한다.  텍스트이(가) 표시된 사진  자동 생성된 설명  문제   * 파일명을 입력받아 "w" 모드로 open한 후 이름과 성적을 입력받아 파일에 저장 이름에 "end"가 입력 되면 file close한 후 'r'모드로 파일을 다시 open 파일에서 자료를 읽어 list에 저장한 후 키보드로부터 검색할 이름을 검색 한 후 있으면 이름과 성적 출력 없으면 "Not foumd error" 출력 파일을 open하는 함수 file\_open()함수 작성   나의 풀이  텍스트이(가) 표시된 사진  자동 생성된 설명  강사님 풀이  텍스트이(가) 표시된 사진  자동 생성된 설명  ->전체적인 알고리즘은 같으나 함수를 사용해서 코드의 가독성과 향후 유지 보수에 더 유리함 |