ILE3-011 디버깅

Debugging @

컴퓨터 프로그램 개발 단계 중에 발생하는 시스템의 논리적인 오류나 비정상적 연산(버그)을 찾아내고 그 원인을 밝히고 수정하는 작업 과정

디버깅 방법 ♂

1. Console 등으로 값을 출력하여 확인

print() 구문을 이용하여 확인하고 싶은 변수를 출력 확인하고 싶은 변수가 바뀔때마다 코드 변경으로 재시작 하여야 함(일부 언어는 실시간 반영 가능)

2. Debugger를 이용하여 in-line Debug

Breakpoint 를 이용하여 확인하고 싶은 위치를 지정하고, 해당 영역에서의 변수 및 함수들에 대한 분석

Visual Studio에서의 Debugging 🔗

- F5 버튼을 눌러 디버깅 형태 설정
- launch.json 이 설정

단축키 🔗

• F5

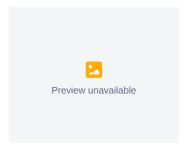
디버깅 시작 or 다음 Breakpoint까지 진행

- F9Breakpoint 추가 / 삭제
- F10

한줄 내려가기

• F11

한줄 내려가기, 함수 일 경우에는 해당 함수 안으로 진입





Preview unavailable

• 변수

Debugging이 수행중인 Line에서의 변수 현황(값 변경 가능)

• 조사식

값 또는 함수의 호출값을 수동으로 확인하기 위하는 기능

• 호출 스택

해당 Line에 진입하기 위해서 진행된 함수 호출 내역

과제 🔗

입력 파일을 읽어서 아래의 코드를 실행 후 오류가 나는 이유를 확인하고 정상적으로 프로그램이 돌아 갈 수 있도록 수정

```
1 def bubbleSort(arr):
     n = len(arr)
3
      swapped = False
4
     for i in range(n-1):
          for j in range(0, n-i-1):
              if arr[j] > arr[j + 1]:
 6
 7
                  swapped = True
8
                  arr[j], arr[j + 1] = arr[j], arr[j+1]
9
          if not swapped:
10
               return
11
```

```
12
13 # Driver code to test above
14 arr = [64, 34, 25, 12, 22, 11, 90]
15
16 bubbleSort(arr)
17
18 print("Sorted array is:")
19 for i in range(len(arr)):
20  print("% d" % arr[i], end=" ")
21
22 # Expected Result
23 # 11 12 22 25 34 64 90
```