

실습 1, 2)

실습1) 2차원리스트 생성

◆ PY03_1_학번_이름.py

◆ 아래 화면처럼 출력되도록 2차원 리스트 생성 프로그램 작성

```
>>>
list_1(0): [1, 2, 3, 4]
list_1(1): [5, 6, 7, 8]
list_1(2): [9, 10, 11, 12]
-----
list_2: [[1, 2, 3, 4], [5, 6, 7, 8], [9, 10, 11, 12]]
-----
list_2[0][0] : 1      list_2[0][1] : 2      list_2[0][2] : 3      list_2[0][3] : 4
list_2[1][0] : 5      list_2[1][1] : 6      list_2[1][2] : 7      list_2[1][3] : 8
list_2[2][0] : 9      list_2[2][1] : 10     list_2[2][2] : 11     list_2[2][3] : 12
>>>
```

실습2) 리스트에서 특정 요소 선택하기

- ◆ PY03_2_학번_이름.py
- ◆ 여러개의 문자열로 구성된 리스트 a (코드 내부에 정의)

```
a = ["alpha", "beta", "gamma", "delta", "epsilon", "zeta"]
```
- ◆ 사용자로부터 선택할 문자열 길이(정수) 입력받음
- ◆ 문자열 중 입력된 숫자 길이의 문자 선택
 - 문자열 길이 함수 : `len(문자열)`
- ◆ 선택된 문자들을 리스트 형태로 출력

```
입력 → 문자열 길이 : 5
출력 → ['alpha', 'gamma', 'delta']
      >>>
```

PY03_3_학번_이름.py

실습 3) 대소문자 변환 프로그램

실습3) 대소문자 변환 프로그램

- ◆ PY03_3_학번_이름.py
- ◆ 아래 화면과 같이 입력 받은 문자열을 대소문자 변환 후 출력
- ◆ 반복적으로 입력받을 수 있도록 구현하고, 'exit' 입력 시 프로그램 종료
- ◆ swapcase() 함수 사용 금지

```
>>>  
문자열 입력 : Hello World  
대소문자 변환 결과 => hELLO wORLD  
문자열 입력 : exit  
>>>
```

PY03_4_학번_이름.py

실습 4) 높은 가격 순으로 출력하기

높은 가격 순으로 출력하기

- ◆ 물품 가격 여러 개가 문자열 한줄로 입력됨
- ◆ 각 가격은 세미콜론(;)으로 구분되어 있음 (중간 공백 없음)
- ◆ 입력예) 61900;83000;16000;250000;7800;196000
- ◆ 입력된 가격을 높은 가격순으로 출력
- ◆ 출력조건)
 - 가격은 전체 길이 9자리
 - 오른쪽 정렬
 - 3자리마다 ,(coma) 추가 (참고) format 함수 사용: format(숫자, ',')

- ◆ 출력예)

250,000
196,000
83,000
61,900
16,000
7,800

PY03_5_학번_이름.py

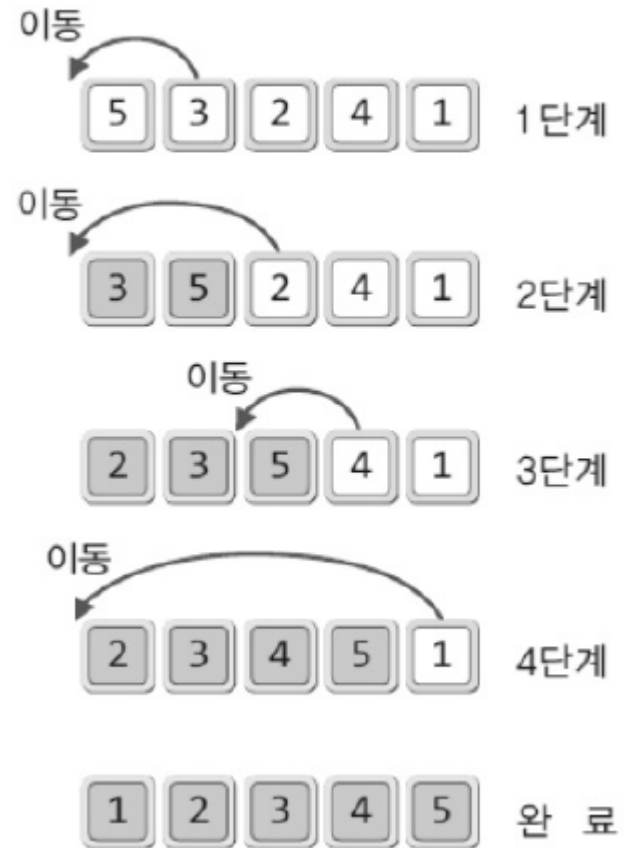
실습 5) 정렬 프로그램 (삽입 정렬)

정렬 (Sorting)

- ◆ 순서 없이 배열된 자료들을 어떤 기준에 따라 오름차순(ascending order)으로 또는 내림차순(descending order)으로 재배열

- ◆ 삽입 정렬

- 원소를 한 번에 하나씩 정해진 순서대로 정렬
- 효율적이지는 않지만 매우 간단
- 입력 데이터의 배열을 저장하기 위해 추가적인 공간(메모리) 불필요



실습5) 삽입정렬 프로그램

◆ 리스트 생성

- 숫자를 입력 받아 리스트에 추가
- '0'을 입력하면 리스트 생성을 멈추고, 정렬 수행

◆ 정렬 수행 단계 중간 리스트 출력

◆ 최종 정렬 완성된 리스트 출력

```
>>>
```

```
입력할 수 : 5  
입력할 수 : 3  
입력할 수 : 2  
입력할 수 : 4  
입력할 수 : 1  
입력할 수 : 0
```

```
-----  
입력된 리스트 : [5, 3, 2, 4, 1]
```

```
1 단계 : [3, 5, 2, 4, 1]
```

```
2 단계 : [2, 3, 5, 4, 1]
```

```
3 단계 : [2, 3, 4, 5, 1]
```

```
4 단계 : [1, 2, 3, 4, 5]
```

```
최종 리스트 : [1, 2, 3, 4, 5]
```

```
-----  
입력할 수 : |
```