



한국IT진흥부설

정보보호교육학원 아이섹

파이썬 기초

데이터 타입(리스트, 튜플)

By SoonGu Hong

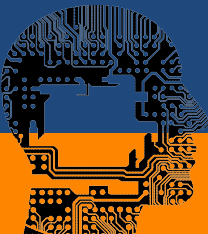


한국IT진흥부설

정보보호교육학원 아이섹

파이썬 기초

1. 리스트



1-1. 리스트란?

- 리스트(list)란 **복수의 데이터를 저장**하기 위한 파이썬의 자료형입니다.
- 리스트를 사용하면 다량의 데이터를 하나의 변수로 관리할 수 있습니다.
- 리스트는 문자열과 같은 **나열 가능한 자료**(sequence)이며 각 데이터에 **인덱스**가 부여됩니다.
- 인덱스를 이용한 읽기, 쓰기를 지원하며 부분 데이터셋을 추출하는 슬라이싱도 문자열과 마찬가지로 지원합니다.

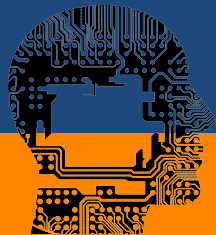


한국IT진흥부설

정보보호교육학원 아이섹

파이썬 기초

2. 리스트 다루기

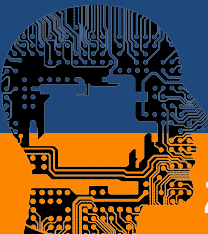




한국IT진흥부설

정보보호교육학원 아이섹

파이썬 기초



2-1. 리스트 인덱싱, 슬라이싱

2-1. 리스트 관련 함수

방법	설명
<code>listData = []</code>	리스트를 만들어 줌
<code>len(listData)</code>	리스트의 항목의 수를 반환
<code>mix(listData), max(listData)</code>	리스트에서 가장 작은(min) 항목과 가장 큰(max) 항목을 반환
<code>listData[start:stop]</code>	리스트의 start 위치부터 stop 위치까지 부분 데이터를 추출(stop 위치의 항목은 포함 안 됨)
<code>listData.append(value)</code>	list에 value를 추가
<code>listData.clear()</code>	list의 모든 항목을 삭제
<code>listData.count(value)</code>	리스트에서 value의 개수를 반환
<code>listData.extend(newList)</code>	list에 newList를 추가
<code>+</code>	두 리스트를 연결함
<code>listData.index(value, position=0)</code>	position위치 이후에서 value의 값이 있는 인덱스를 반환
<code>listData.insert(index, value)</code>	list의 index위치에 value를 삽입
<code>listData.remove(value)</code>	리스트에서 해당 값을 삭제
<code>del listData[index]</code>	리스트에서 인덱스를 이용해 항목을 삭제
<code>listData.pop()</code>	리스트에서 가장 마지막 항목을 반환하고 삭제
<code>listData.reverse()</code>	리스트의 항목들의 순서를 반대로 함
<code>listData.sort(reverse=False)</code>	리스트의 항목들을 정렬. reverse 속성을 True로 하면 내림차순으로 정렬

2-2. 리스트 인덱싱

```
idol = ['EXO', 'BTS', 'IZONE', 'NCT127', 'APINK']
```

EXO	BTS	IZONE	NCT127	APINK
0	1	2	3	4
-5	-4	-3	-2	-1

```
idol[2] -> "IZONE"  
idol[3][1] -> "C"  
idol[4][1:3] -> "PI"
```

```
idol[2] = "IOI"  
-> IZONE이 IOI로  
수정됨!
```

2-3. 리스트 슬라이싱

```
idol = ['EXO', 'BTS', 'IZONE', 'NCT127', 'APINK']
```

EXO	BTS	IZONE	NCT127	APINK
0	1	2	3	4
-5	-4	-3	-2	-1

리스트데이터[begin:end:step] - begin포함 end미포함

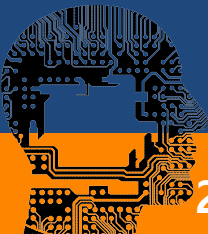
ex) idol[2:4:1] -> ["IZONE", "NCT127"]



한국IT진흥부설

정보보호교육학원 아이섹

파이썬 기초



2-2. 리스트 데이터 추가/삭제

2-2-1. 주요함수

1. `append()` - 데이터를 리스트의 맨 끝에 추가
2. `insert()` - 데이터를 리스트의 특정 위치에 삽입
3. `remove()` - 리스트의 데이터를 값을 통해 삭제
4. `del()` - 리스트의 데이터를 인덱스를 통해 삭제
5. `clear()` - 리스트 내부 데이터 전체 삭제

< 예시 리스트 `nums` >

```
nums = [1, 3, 5, 7, 9, 11, 13]
```

2-2-2. 사용 예시

```
nums.append(13)
```

```
# insert(index, value)
```

```
nums.insert(2, 4)
```

```
nums.remove(9) # 9 삭제
```

```
del(nums[0]) # 0번인덱스 삭제
```

```
nums.clear()
```



```
[1, 3, 4, 5, 7, 9, 11, 13]
```

```
[1, 3, 4, 5, 7, 11, 13]
```

```
[3, 4, 5, 7, 11, 13]
```

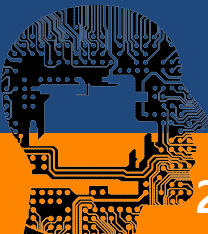
```
[]
```



한국IT진흥부설

정보보호교육학원 아이섹

파이썬 기초



2-3. 리스트 데이터 탐색/정렬

2-3-1. 주요함수

1. `index()`: 리스트에서 특정 값이 저장된 인덱스를 반환.
2. `count()`: 리스트 내부에 저장된 특정 데이터의 개수를 반환.
3. `sort()`: 리스트를 오름차 정렬.
4. `reverse()`: 리스트 데이터를 역순으로 정렬.

< 예시 리스트 `points` >

```
points = [99, 14, 87, 100, 55, 100, 99, 100, 22]
```

2-3-2. 사용 예시1

```
points = [99, 14, 87, 100, 55, 100, 99, 100, 22]

perfect = points.count(100)
print("만점자의 수는 %d명입니다." % perfect)

idx = points.index(87)
print("87이 저장된 위치는 %d번입니다." % idx)

# 내장함수 len(), max(), min()
print("학생 수는 %d명입니다." % len(points))
print("최고점수는 %d점입니다." % max(points))
print("최저점수는 %d점입니다." % min(points))
```

```
만점자의 수는 3명입니다.
87이 저장된 위치는 2번입니다.
학생 수는 9명입니다.
최고점수는 100점입니다.
최저점수는 14점입니다.
```

2-3-3. 사용 예시2

```
# 오름차 정렬 sort
print("-" * 40)
print(points)
points.sort()
print(points)

points.reverse()
print(points)
```

```
[99, 14, 87, 100, 55, 100, 99, 100, 22]
[14, 22, 55, 87, 99, 99, 100, 100, 100]
[100, 100, 100, 99, 99, 87, 55, 22, 14]
```

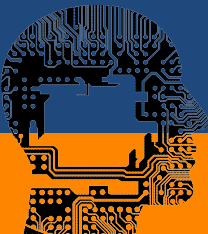


한국IT진흥부설

정보보호교육학원 아이섹

파이썬 기초

3. 튜플



3-1. 튜플이란?

- 튜플(tuple)은 값의 집합이라는 측면에서 리스트와 유사하지만 값을 한 번 저장한 후에는 내부 요소의 편집이 불가능합니다.
- 튜플은 상수리스트라고도 부르며 ()를 사용하여 표현합니다.
- 튜플은 값의 집합만 표현할 뿐 바뀔 일이 없으므로 내부구조가 단순하고 속도도 빠릅니다.
- 리스트와 튜플은 값 변경 가능성 여부만 다를 뿐 구조가 비슷해 상호변경이 가능합니다.

감사합니다
THANK YOU