

## 5. 해시와 데이터베이스: 데이터 제자리에 넣기



순천향대학교 컴퓨터공학과  
이 상 정

순천향대학교 컴퓨터공학과

1

프로그래밍 기초

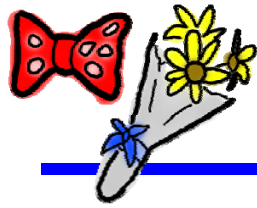
### 학습 내용

- 배열이 데이터를 저장하는 유일한 장소는 아님
- 해시(hash) 또는 딕셔너리(dictionary, 사전)
  - 점수와 이름을 연관시켜 저장
  - 키(key):값(value) 저장
- 데이터베이스(database)
  - 디스크에 저장되어 있는 데이터로 작업 시 외부 데이터베이스 시스템으로부터 읽을 수 있음

순천향대학교 컴퓨터공학과

2

5. 해시와 데이터베이스



# 누가 서핑경기 우승자?

- 4장의 서핑대회 코드는 상위 3명의 점수만 출력
  - 누가 우승자?
  
- 두 개 배열 사용해서 점수 출력
  - 한 배열은 점수
  - 또 다른 배열은 선수 이름 저장



# 서핑대회 코드- 이름과 점수 출력1 [실습 1]



```

scores = []
names = []

result_f = open("results.txt")

for line in result_f:
    (name, score) = line.split()
    scores.append(float(score))
    names.append(name)

result_f.close()

scores.sort()
scores.reverse()

```

```

names.sort()
names.reverse()

print("The highest scores were:")

print(names[0]+' with '+str(scores[0]))
print(names[1]+' with '+str(scores[1]))
print(names[2]+' with '+str(scores[2]))

```



```

Python Shell
File Edit Shell Debug Options Windows Help
Python 3.2.2 (default, Sep 4 2011, 09:51:08) [MSC v.1500 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ===== RESTART =====
>>>
The highest scores were:
Zack with 9.12
Stacey with 8.65
Juan with 8.45
>>> |
Ln: 9 Col: 4

```



❑ Zack은 초보자인데...

Jonny	8.65
Juan	9.12
Joseph	8.45
Stacey	7.81
Aideen	8.05
Zack	7.21
Aaron	8.31
.	
.	

## 이름과 점수를 연결 - 해시

❑ 두 개의 배열이 서로 연관 없이 정렬되어 제대로 작동되지 않음

- 이름과 점수 간의 연결을 유지할 수 있는 자료구조 사용
- 파이썬의 해시 를 사용하여 이름과 점수 연결

❑ 해시 예

- `h = {}` # 빈 해시 생성
- `h['one'] = 1` # 대괄호 안의 값(one)은 키(key), 1은 값(value)
- `h['two'] = 2`
- `h = {'one':1, 'two':2 }`
- `scores = {}`
- `scores[8.45] = 'Joseph'`

```

Python 3.4.2 Shell
File Edit Shell Debug Options Windows Help
Python 3.4.2 (v3.4.2:ab2c023a9432, Oct 6 2014) on win32
Type "copyright", "credits" or "license()"
>>>
>>> h = { 'one':1, 'two':2 }
>>> h
{'one': 1, 'two': 2}
>>> h['셋']=3
>>> h
{'셋': 3, 'one': 1, 'two': 2}
>>> |
    
```

## □ 뷰(view)

- 해시의 항목을 동적으로 볼 수 있는 객체
- `keys()` : 키에 대한 해시 뷰
- `values()` : 값에 대한 해시 뷰
- `items()` : 항목들의 해시 뷰

## □ 해시 뷰 예

- `h = {'one':1, 'two':2 }`  
`h.keys()`
- `scores = {}` # 빈 해시 생성  
`scores[8.45] = 'Joseph'` # 대괄호 안의 값(점수)은 키(key)  
`for key in scores.keys()` # `keys()` 메서드는 해시의 키들의 배열 반환  
`print(scores[key] + ' had a score of ' + key)`
- `for score, surfer in scores.items()` # `items()` 메서드는 모든 키-값 쌍을 반환  
`print(surfer + ' had a score of ' + str(score))`

# 해시 뷰 예

```
Python 3.4.2 (v3.4.2:ab2c023a9432, Oct 6 2014, 22
tel)] on win32
Type "copyright", "credits" or "license()" for mor
>>>
>>> h = { 'one':1, 'two':2 }
>>> h
{'one': 1, 'two': 2}
>>> h['셋']=3
>>> h
{'셋': 3, 'one': 1, 'two': 2}
>>>
>>> h.keys()
dict_keys(['셋', 'one', 'two'])
>>> h.values()
dict_values([3, 1, 2])
>>> h.items()
dict_items([('셋', 3), ('one', 1), ('two', 2)])
>>>
>>> for key in h.keys():
        print(key)

셋
one
two
>>> |
```

# 서핑대회 코드- 이름과 점수 출력2 [실습 2]



```
scores = {}
result_f = open("results.txt")

for line in result_f:
    (name, score) = line.split()
    scores[score] = name

result_f.close()

print("The top scores were:")
for each_score in scores.keys():
    print('Surfer ' + scores[each_score] + ' scored ' + each_score)
```



```
Python Shell
File Edit Shell Debug Options Windows Help
Python 3.2.2 (default, Sep  4 2011, 09:51:08) [MSC v.1500 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ----- RESTART -----
>>>
The top scores were:
Surfer Joseph scored 8.45
Surfer Juan scored 9.12
Surfer Zack scored 7.21
Surfer Aaron scored 8.31
Surfer Aideen scored 8.05
Surfer Johnny scored 8.65
Surfer Stacey scored 7.81
>>> |
```

❑ 데이터가 정렬이 되지 않았네.....



❑ 파이선 해시는 `sorted()` 함수를 사용하여 키를 정렬

- 예  
`sorted(scores.keys(), reverse = True)`
- 예  
`for each_score in sorted(scores.keys(), reverse = True):`  
`print('Surfer ' + scores[each_score] + ' scored ' + each_score)`

# 서핑대회 코드- 이름과 점수 정렬 출력 [실습 3]



```

scores = {}
result_f = open("results.txt")

for line in result_f:
    (name, score) = line.split()
    scores[score] = name

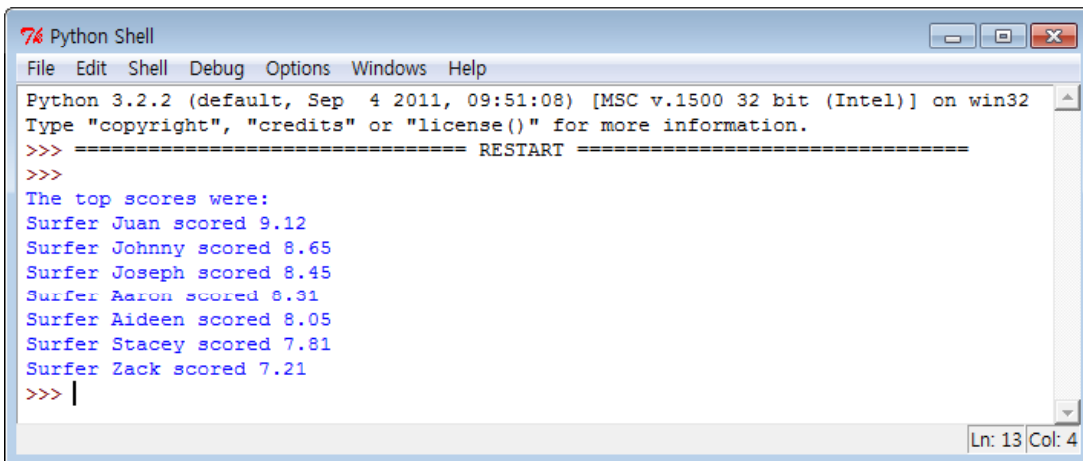
result_f.close()

print("The top scores were:")
for each_score in sorted(scores.keys(), reverse = True):
    print('Surfer ' + scores[each_score] + ' scored ' + each_score)

```



# 시험주행



## 복잡한 데이터 처리 요청

- 지역 서핑 연합회(지서연)에서 프로그램 개발 요청
  - 지서연의 데이터는 전용 데이터 형식으로 텍스트 파일 저장
  - 각 선수 별로 여섯 개의 데이터 항목이 저장  
ID, Name, Country, Average, Board type, Age

101;Johnny 'wave-boy' Jones;USA;8.32;Fish;21

```
ID:          101
Name:        Johnny 'wave-boy' Jones
Country:     USA
Average:     8.32
Board type:  Fish
Age:         21
```



## 지서연 서핑대회 코드- 복잡한 데이터 출력

```
line = "101;Johnny 'wave-boy' Jones;USA;8.32;Fish;21"

s = {}
(s['id'], s['name'], s['country'], s['average'], s['board'], s['age']) = line.split(";")

print("ID:          " + s['id'])
print("Name:        " + s['name'])
print("Country:     " + s['country'])
print("Average:     " + s['average'])
print("Board type:  " + s['board'])
print("Age:         " + s['age'])
```





```

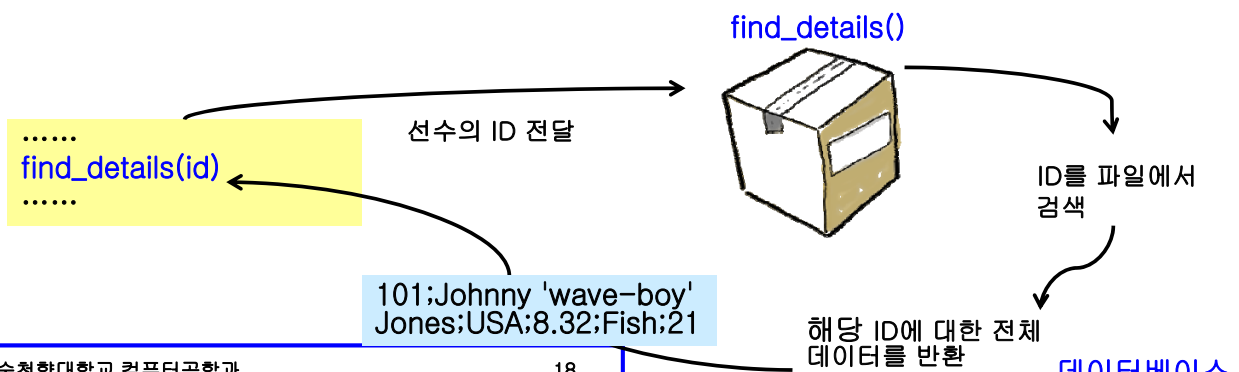
Python Shell
File Edit Shell Debug Options Windows Help
Python 3.2.2 (default, Sep 4 2011, 09:51:08) [MSC v.1500 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ===== RESTART =====
>>>
ID:          101
Name:        Johnny 'wave-boy' Jones
Country:     USA
Average:     8.32
Board type:  Fish
Age:         21
>>> |

```

이젠 한 선수가 아닌 여러 선수의 데이터를 출력

## 함수에서 구조화된 데이터 반환

- 단 한 선수가 아닌 전체 파일에서 데이터를 처리
- 프로그램이 특정 선수에 대한 정보 출력을 요청 받으면 가능한 한 빨리 해당 데이터를 가져오는 함수 작성
  - 선수의 ID를 입력
  - 해당 선수의 데이터가 나올 때까지(ID가 일치할 때까지) 파일에서 데이터를 검색
  - 검색된 데이터를 해시로 반환



```
def find_details(id2find): # 함수 정의
    surfers_f = open("surfing_data.csv");
    for eachline in surfers_f:
        s = {}
        (s['id'], s['name'], s['country'], s['average'], s['board'], s['age']) = eachline.split(";")
        if id2find == int(s['id']): # ID 검색, 찾았으면
            surfers_f.close()
            return(s) # 해시 리턴
    surfers_f.close()
    return({}) # 없으면 빈 해시 리턴
```

```
lookup_id = int(input("Enter the id of the surfer: "))
surfer = find_details(lookup_id)
if surfer:
    print("ID: " + surfer['id'])
    print("Name: " + surfer['name'])
    print("Country: " + surfer['country'])
    print("Average: " + surfer['average'])
    print("Board type: " + surfer['board'])
    print("Age: " + surfer['age'])
```



```
Python Shell
File Edit Shell Debug Options Windows Help
Python 3.2.2 (default, Sep 4 2011, 09:51:08) [MSC v.1500 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ----- RESTART -----
>>>
Enter the id of the surfer: 104
ID: 104
Name: Stacey O'Neill
Country: Ireland
Average: 8.91
Board type: Malibu
Age: 22
>>> |
```



# 데이터베이스 검색

- ❑ 헤드 퍼스트 TVN 수상 스포츠 전문 방송국으로 전국 서핑 대회 중계
  - 지서연 프로그램과 동일하게 선수의 정보 검색 원함
  - TVN 데이터는 파일이 아니라 **데이터베이스**에 저장
    - **데이터베이스 시스템**은 대용량의 데이터를 효율적으로 처리(검색, 삽입, 삭제 등)하고 저장하는 시스템
    - 데이터베이스의 데이터는 텍스트 파일이 아닌 특수한 형식의 데이터베이스 파일에 데이터를 저장
  - TVN은 **SQLite3** 데이터베이스를 사용
- ❑ 파이썬은 데이터베이스와 연결하여 처리하는 기능 지원
- ❑ 9장에서 **데이터베이스 활용** 소개



## TVN 서핑대회 코드- 데이터베이스 연결 (1) [실습 5]

```
import sqlite3                # SQLite3 라이브러리 임포트

def find_details(id2find):
    db = sqlite3.connect("surfersDB.sdb") # 데이터베이스 연결
    db.row_factory = sqlite3.Row         # 열 이름으로 참조
    cursor = db.cursor()
    cursor.execute("select * from surfers") # 테이블의 모든 행 읽음
    rows = cursor.fetchall()             # 읽은 행을 rows에 저장

    for row in rows:
        if row['id'] == id2find:
            s = {}
            s['id'] = str(row['id'])
            s['name'] = row['name']
            s['country'] = row['country']
            s['average'] = str(row['average'])
            s['board'] = row['board']
```



# TVN 서핑대회 코드: 데이터베이스 연결 (2)

```

s['age'] = str(row['age'])
cursor.close()
return(s)
cursor.close()
return({})

lookup_id = int(input("Enter the id of the surfer: "))
surfer = find_details(lookup_id)
if surfer:
    print("ID:          " + surfer['id'])
    print("Name:         " + surfer['name'])
    print("Country:       " + surfer['country'])
    print("Average:       " + surfer['average'])
    print("Board type:    " + surfer['board'])
    print("Age:          " + surfer['age'])

```



# 시험주행

```

Python Shell
File Edit Shell Debug Options Windows Help
Python 3.2.2 (default, Sep 4 2011, 09:51:08) [MSC v.1500 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ===== RESTART =====
>>>
Enter the id of the surfer: 102
ID:          102
Name:         Juan Martino
Country:       Spain
Average:       9.01
Board type:    Gun
Age:          36
>>> ===== RESTART =====
>>>
Enter the id of the surfer: 298
>>> ===== RESTART =====
>>>
Enter the id of the surfer: 106
ID:          106
Name:         Zack 'bonnie-lad' MacFadden
Country:       Scotland
Average:       7.82
Board type:    Thruster
Age:          26
>>> |

```





## 프로그래밍 도구상자 (1)

### □ 프로그래밍 도구

- **해시** - 이름과 값을 연관시키는 데이터 구조체
- **s['age']** - s 해시에서 age라는 이름에 연관되어 있는 값을 가져옴
- 함수에서 자료 구조도 반환
- **데이터베이스시스템** - SQLite3와 같이 대용량의 데이터를 매우 효율적으로 처리하고 저장하는 기술



## 프로그래밍 도구상자 (2)

### □ 파이썬 도구

- **{}** - 빈 해시
- **s['wind'] = "off shore"** - s 해시 안에서 wind와 연관된 값을 'off shore'로 설정
- **s.keys()** - s 해시에 들어 있는 키의 리스트를 반환
- **s.values()** - s 해시에 들어 있는 값의 리스트를 반환
- **s.items()** - s 해시에 들어 있는 키와 값의 리스트를 반환
- **line.split(",")** - line 변수 안에 들어 있는 스트링을 콤마가 나올 때 마다 분할
- **sorted()** - 대부분의 데이터 구조체를 정렬할 수 있는 내장 함수

## 5장 실습

[실습 1] 서핑대회 코드- 이름과 점수 출력1

[실습 2] 서핑대회 코드- 이름과 점수 출력2

[실습 3] 서핑대회 코드- 이름과 점수 정렬 출력

[실습 4] 지서연 서핑대회 코드- 선수 검색 출력

[실습 5] TVN 서핑대회 코드- 데이터베이스 연결

[실습 6] 파이썬 매뉴얼의 Tutorial의 Dictionaries 예제 실행



## 5장 과제

1. TVN 서핑대회 코드를 수정하여 한 선수 검색 후 종료하지 않고 반복해서 수행하도록 수정
  - ID를 -1로 입력하면 프로그램 종료
2. 앞에서 배운 내용을 사용한 임의의 프로그램 작성
  - 프로그램 설명
  - 프로그램 소스
  - 실행 결과