

CSE 160 Section 7 Solutions

1. {1, 6, 3, 9, 5, 2, 4} (note values in random order)

2.

- `print(set_one & set_two)` (The intersection is: {'a', 'c', 'd'})
- `print(set_one | set_two)` (The union is: {'a', 'b', 'c', 'd', 'e', 'f', 'g'})
- `print(set_one - set_two)` (The difference is: {'b', 'e', 'f'})
- `print(set_two - set_one)` (The difference is: {'g'})

3.

```
def all_unique_words(file_name):
    f = open(file_name, "r") # "r" is optional
    unique_words = set()
    for line in f:
        words = line.split()
        for word in words:
            unique_words.add(word)
    return len(unique_words)
```

4.

```
[('Sam', 5, 6), ('Ann', 7, 8), ('Fred', 3, 5), ('Zeke', 5, 3),
('Mary', 3, 5)]
```

```
[('Ann', 7, 8), ('Sam', 5, 6), ('Fred', 3, 5), ('Mary', 3, 5),
('Zeke', 5, 3)]
```

5.

- `alphabetical_lst = sorted(lst, key = itemgetter(0))`
- `youngest_to_oldest = sorted(lst, key = itemgetter(1))`
-

```
from operator import itemgetter
```

```
def find_oldest(age_list):
    sort_name = sorted(age_list, key=itemgetter(0))
    sort_age = sorted(sort_name, key=itemgetter(1), reverse=True)
    oldest_age = sort_age[0][1]
    ret_list = []
```

```

for pair in sort_age:
    if(pair[1] == oldest_age):
        ret_list.append(pair)
    else:
        return ret_list # return early since it is sorted
return ret_list

```

```

6. def two_stops_away(graph, stop):
    result = set()
    next_stops = get_next_stops(graph, stop)
    for next_stop in next_stops:
        next_next_stops = get_next_stops(graph, next_stop)
        result = result | next_next_stops

    # remove the given stop
    result.remove(stop)

    return result

```

7.

a. The code tries to mutate a tuple, but tuples are immutable - the values stored in a tuple cannot be changed.

b. **One possible solution:**

```

def likes_per_user(yips):
    likes = {}
    for yip in yips:
        name = yip['name']
        if yip['name'] not in likes:
            likes[name] = 0
        likes[name] = likes[name] + yip['likes']
    result = []
    for name in likes:
        result.append((name, likes[name]))
    return result

yips = get_yips()

```

```
likes = likes_per_user(yips)
print(likes)
```