Assignment 4_alt

April 12, 2025

Assignment 4 revised 2/18/2025 ### Beija Richardson ### 4/12/2025

0.1 Question 1

1) import the random library.
2) Use `random.seed(10)` to initialize a pseudorandom number generator.
3) Create a list of 50 random integers from 0 to 15. Call this list `int_list`.
4) Print the 10th and 30th elements of the list.

You will need to use list comprehension to do this. The syntax for list comprehension is: = `[<expression> for <item> in <iterable>]`. For this question your expression will be a randint generator from the random library and your iterable will be `range()`. Researh the documentation on how to use both functions.

[7]:
```python
import random
random.seed(10)
int_list = [random.randint(0, 15) for _ in range(50)]
print("10th element:", int_list[9])
print("30th element:", int_list[29])
```

```
10th element: 1
30th element: 7
```

0.2 Question 2

1) import the string library.
2) Create the string `az_upper` using `string.ascii_uppercase`. This is a single string of uppercase letters
3) Create a list of each individual letter from the string. To do this you will need to iterate over the string and append each letter to the an empty list. Call this list `az_list`.
4) Print the list.

You will need to use a for-loop for this. The syntax for this for-loop should be:

```
for i in string>:    <list operation>
```

[9]:
```
import string
az_upper = string.ascii_uppercase
az_list = []
for letter in az_upper:
    az_list.append(letter)
print(az_list)
```

```
['A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O', 'P',
'Q', 'R', 'S', 'T', 'U', 'V', 'W', 'X', 'Y', 'Z']
```

0.3 Question 3

1) Create a set from 1 to 5. Call this `set_1`.
2) Create a set from int_list. Call this `set_2`.
3) Create a set by finding the `symmetric_difference()` of `set_1` and `set_2`. Call this `set_3`.
4) What is the length of all three sets?

[13]:
```
set_1 = set(range(1, 6))
set_2 = set(int_list)
set_3 = set_1.symmetric_difference(set_2)
print("Length of set_1:", len(set_1))
print("Length of set_2:", len(set_2))
print("Length of set_3:", len(set_3))
```

```
Length of set_1: 5
Length of set_2: 15
Length of set_3: 12
```

0.4 Question 4

Complete exercise 9.15.3 from Think Python by Downey

https://allendowney.github.io/ThinkPython/chap09.html

[19]:
```
reversed('parrot')
list(reversed('parrot'))
''.join(reversed('parrot'))
```

[19]:
```
'torrap'
```

[21]:
```
def reverse_word(word):
    return ''.join(reversed(word))
```

[35]:
```
def is_palindrome(word):
    return word == reverse_word(word)
word_list = ['blue','cargo','happy','tree','rollercoaster','cake']
for word in word_list:
    if len(word) >= 7 and is_palindrome(word):
        print(word)
```

[53]:
```
# 3.1

%load_ext sql
%sql sqlite:///bank.db

%%sql
SELECT emp_id, fname, lname
FROM employee
ORDER BY lname, fname;
```

```
  Cell In[53], line 7
    SELECT emp_id, fname, lname
           ^
SyntaxError: invalid syntax
```

[41]:
```
# 3.2
SELECT account_id, cust_id, avail_balance
FROM account
WHERE status = 'ACTIVE'
  AND avail_balance > 2500;
```

```
  Cell In[41], line 2
    SELECT account_id, cust_id, avail_balance
           ^
SyntaxError: invalid syntax
```

[ ]:
```
# 3.3
SELECT DISTINCT open_emp_id
FROM account;
```

[ ]:
```
# 3.4
SELECT p.product_cd, a.cust_id, a.avail_balance
FROM product p
         INNER JOIN account a
                    ON p.product_cd = a.product_cd
WHERE p.product_type_cd = 'ACCOUNT';
```