Additional Test cases to see if your solution is passing the test cases

FindBusinessBasedOnCity

Test Case 1:

```
true_results =['3 Palms$7707 E McDowell Rd, Scottsdale, AZ 85257$Scottsdale$AZ', "Bob's Bike Shop$1608 N Miller Rd, Scottsdale, AZ 85257$Scottsdale$AZ", 'Ronan & Tagart, PLC$8980 E Raintree Dr, Ste 120, Scottsdale, AZ 85260$Scottsdale$AZ', "Sangria's$7700 E McCormick Pkwy, Scottsdale, AZ 85258$Scottsdale$AZ", 'Turf Direct$8350 E Evans Rd, Scottsdale, AZ 85260$Scottsdale$AZ']

try:

FindBusinessBasedOnCity('Scottsdale', 'output_city.txt', data)

except NameError as e:

print ('The FindBusinessBasedOnCity function is not defined! You must run the cell containing the function before running this evaluation cell.')

except TypeError as e:

print(e)

print ("The FindBusinessBasedOnCity function is supposed to accept three arguments. Yours does not!")

try:

opf = open('output_city.txt', 'r')

except FileNotFoundError as e:

print ("The FindBusinessBasedOnCity function does not write data to the correct location.")

lines = opf.readlines()

if len(lines) != 5:

print ("The FindBusinessBasedOnCity function does not find the correct number of results, should be 3.")

lines = [line.strip() for line in lines]

if sorted(lines) == sorted(true_results):
```

print ("Correct! You FindBusinessByCity function passes these test cases. This does not cover all possible test edge cases, however, so make sure that your function covers them before submitting!")

Test Case 2:

true_results =['Arizona Exterminating Co.$521 E Broadway Rd, Mesa, AZ 85204$Mesa$AZ', 'Bikram Yoga$1940 W 8th St, Ste 111, Mesa, AZ 85202$Mesa$AZ', "Denny's Restaurant$1330 S Power Rd, Mesa, AZ 85206$Mesa$AZ", 'Diamondback Gymnastics$7211 E Southern Avenue, Mesa, AZ 85209$Mesa$AZ', 'Southeast Valley Medical Group$1950 S Country Club Dr, Mesa, AZ 85210$Mesa$AZ', 'Spa Pima$2150 S Power Rd, Mesa, AZ 85209$Mesa$AZ', 'The Seafood Market$1910 S Gilbert Rd, Mesa, AZ 85204$Mesa$AZ']

try:

FindBusinessBasedOnCity('Mesa', 'output_city.txt', data)

except NameError as e:

print ('The FindBusinessBasedOnCity function is not defined! You must run the cell containing the function before running this evaluation cell.')

except TypeError as e:

print(e)

print ("The FindBusinessBasedOnCity function is supposed to accept three arguments. Yours does not!")

try:

opf = open('output_city.txt', 'r')

except FileNotFoundError as e:

print ("The FindBusinessBasedOnCity function does not write data to the correct location.")

lines = opf.readlines()

if len(lines) != 7:

print ("The FindBusinessBasedOnCity function does not find the correct number of results, should be 3.")

```
lines = [line.strip() for line in lines]

if sorted(lines) == sorted(true_results):

print ("Correct! You FindBusinessByCity function passes these test cases. This does not cover all possible test edge cases, however, so make sure that your function covers them before submitting!")
```

FindBusinessBasedOnLocation

Test Case 1:

```
true_results =['Turf Direct']

try:

FindBusinessBasedOnLocation(['Gardeners'], [33.3482589, -111.9088346], 20, 'output_loc.txt', data)

except NameError as e:

print ('The FindBusinessBasedOnLocation function is not defined! You must run the cell containing the function before running this evaluation cell.')

except TypeError as e:

print ("The FindBusinessBasedOnLocation function is supposed to accept five arguments. Yours does not!")

try:

opf = open('output_loc.txt','r')

except FileNotFoundError as e:

print ("The FindBusinessBasedOnLocation function does not write data to the correct location.")

lines = opf.readlines()

if len(lines) != 1:

print ("The FindBusinessBasedOnLocation function does not find the correct number of results, should be only 1.")

lines = [line.strip() for line in lines]
```

```
if sorted(lines) == sorted(true_results):

print ("Correct! Your FindBusinessBasedOnLocation function passes these test cases. This does not cover all possible edge cases, so make sure your function does before submitting.")
```

Test Case 2:

```
true_results = ['Nothing Bundt Cakes', 'P.croissants']

try:

FindBusinessBasedOnLocation(['Bakeries'], [33.3482589, -111.9088346], 15, 'output_loc.txt', data)


except NameError as e:

print ('The FindBusinessBasedOnLocation function is not defined! You must run the cell containing the function before running this evaluation cell.')

except TypeError as e:

print ("The FindBusinessBasedOnLocation function is supposed to accept five arguments. Yours does not!")

try:

opf = open('output_loc.txt','r')

except FileNotFoundError as e:

print ("The FindBusinessBasedOnLocation function does not write data to the correct location.")

lines = opf.readlines()

if len(lines) != 2:

print ("The FindBusinessBasedOnLocation function does not find the correct number of results, should be only 1.")

lines = [line.strip() for line in lines]

if sorted(lines) == sorted(true_results):
```

print ("Correct! Your FindBusinessBasedOnLocation function passes these test cases. This does not cover all possible edge cases, so make sure your function does before submitting.")

Test Case 3:

true_results = ['Nothing Bundt Cakes', 'Olive Creations', 'P.croissants', 'The Seafood Market']

try:

FindBusinessBasedOnLocation(['Food', 'Specialty Food'], [33.3482589, -111.9088346], 30, 'output_loc.txt', data)

except NameError as e:

print ('The FindBusinessBasedOnLocation function is not defined! You must run the cell containing the function before running this evaluation cell.')

except TypeError as e:

print ("The FindBusinessBasedOnLocation function is supposed to accept five arguments. Yours does not!")

try:

opf = open('output_loc.txt','r')

except FileNotFoundError as e:

print ("The FindBusinessBasedOnLocation function does not write data to the correct location.")

lines = opf.readlines()

if len(lines) != 4:

print ("The FindBusinessBasedOnLocation function does not find the correct number of results, should be only 1.")

lines = [line.strip() for line in lines]

if sorted(lines) == sorted(true_results):

print ("Correct! Your FindBusinessBasedOnLocation function passes these test cases. This does not cover all possible edge cases, so make sure your function does before submitting.")