



course_4_assessment_4

Due: 2019-02-04 15:16:00

Description: Assessment for the Exceptions lesson

Score: 0 of 8 = 0.0%

Questions

Not yet graded

The code below takes the list of country, `country`, and searches to see if it is in the dictionary `gold` which shows some countries who won gold during the Olympics. However, this code currently does not work. Correctly add try/except clause in the code so that it will correctly populate the list, `country_gold`, with either the number of golds won or the string "Did not get gold".

Save & Run

5/15/2021, 3:15:03 PM - 2 of 2

Show in CodeLens

```
1 gold = {"US":46, "Fiji":1, "Great Britain":27, "Cuba":5, "Thailand":2, "China":26, "France"
2 country = ["Fiji", "Chile", "Mexico", "France", "Norway", "US"]
3 country_gold = []
4
5 for x in country:
6     try:
7         country_gold.append(gold[x])
8     except KeyError:
9         country_gold.append("Did not get gold")
10
11
```

ActiveCode (ac_exceptions_01)			
Result	Actual Value	Expected Value	Notes
Pass	[1, ..., 46]	[1, ..., 46]	Testing that country_gold is assigned to correct values

You passed: 100.0% of the tests

Not yet graded

Provided is a buggy for loop that tries to accumulate some values out of some dictionaries. Insert a try/except so that the code passes.

Save & Run

5/15/2021, 3:15:20 PM - 2 of 2

Show in CodeLens

```
1 di = [{"Puppies": 17, "Kittens": 9, "Birds": 23, "Fish": 90, "Hamsters": 49}, {"Puppies":
2 total = 0
3 for diction in di:
4     try:
5         total = total + diction['Puppies']
6     except KeyError:
7         continue
8
9 print("Total number of puppies:", total)
10
11
```

Total number of puppies: 130

ActiveCode (ac_exceptions_011)			
Result	Actual Value	Expected Value	Notes
Pass	130	130	Testing that total has the correct value.

You passed: 100.0% of the tests

Not yet graded

The list, `numb`, contains integers. Write code that populates the list `remainder` with the remainder of 36 divided by each number in `numb`. For example, the first element should be 0, because 36/6 has no remainder. If there is an error, have the string "Error" appear in the `remainder`.

Save & Run

5/15/2021, 3:15:36 PM - 2 of 2

Show in CodeLens

```
1
2 numb = [6, 0, 36, 8, 2, 36, 0, 12, 60, 0, 45, 0, 3, 23]
3
4 remainder = []
```

```

5
6 for n in numb:
7     try:
8         remainder.append(36%n)
9     except ZeroDivisionError:
10        remainder.append("Error")
11
12 print(remainder)

```

[0, 'Error', 0, 4, 0, 0, 'Error', 0, 36, 'Error', 36, 'Error', 0, 13]

ActiveCode (ac_exceptions_02)

Result	Actual Value	Expected Value	Notes
Pass	[0,..., 13]	[0,..., 13]	Testing that remainder is assigned to correct values.

You passed: 100.0% of the tests

Not yet graded

Provided is buggy code, insert a try/except so that the code passes.

Save & Run

5/15/2021, 3:15:54 PM - 2 of 2

Show in CodeLens

```

1
2 lst = [2,4,10,42,12,0,4,7,21,4,83,8,5,6,8,234,5,6,523,42,34,0,234,1,435,465,56,7,3,43,23]
3
4 lst_three = []
5
6 for num in lst:
7     try:
8         if 3 % num == 0:
9             lst_three.append(num)
10    except ZeroDivisionError:
11        continue
12

```

ActiveCode (ac_exceptions_021)

Result	Actual Value	Expected Value	Notes
Pass	[1, 3]	[1, 3]	Testing that lst_three has the correct values.

You passed: 100.0% of the tests

Not yet graded

Write code so that the buggy code provided works using a try/except. When the code does not work in the try, have it append to the list `attempt` the string "Error".

Save & Run

5/15/2021, 3:16:10 PM - 2 of 2

Show in CodeLens

```

1
2 full_lst = ["ab", 'cde', 'fgh', 'i', 'jkml', 'nop', 'qr', 's', 'tv', 'wxy', 'z']
3
4 attempt = []
5
6 for elem in full_lst:
7     try:
8         attempt.append(elem[1])
9     except IndexError:
10        attempt.append("Error")
11
12

```

ActiveCode (ac_exceptions_03)

Result	Actual Value	Expected Value	Notes
Pass	['b'...ror']	['b'...ror']	Testing that attempt has the correct values.

You passed: 100.0% of the tests

Not yet graded

The following code tries to append the third element of each list in `conts` to the new list `third_countries`. Currently, the code does not work. Add a try/except clause so the code runs without errors, and the string 'Continent does not have 3 countries' is appended to `countries` instead of producing an error.

Save & Run 5/15/2021, 3:16:26 PM - 2 of 2 Show in CodeLens

```

1
2 conts = [['Spain', 'France', 'Greece', 'Portugal', 'Romania', 'Germany'], ['USA', 'Mexico']
3
4 third_countries = []
5
6 for c in conts:
7     try:
8         third_countries.append(c[2])
9     except IndexError:
10        third_countries.append("Continent does not have 3 countries")
11

```

ActiveCode (ac_exceptions_031)

Result	Actual Value	Expected Value	Notes
Pass	['Gre...ies']	['Gre...ies']	Testing that third_countries is created correctly.

You passed: 100.0% of the tests

Not yet graded

The buggy code below prints out the value of the sport in the list `sport`. Use try/except so that the code will run properly. If the sport is not in the dictionary, `ppl_play`, add it in with the value of 1.

Save & Run 5/15/2021, 3:16:40 PM - 2 of 2 Show in CodeLens

```

1
2 sport = ["hockey", "basketball", "soccer", "tennis", "football", "baseball"]
3
4 ppl_play = {"hockey":4, "soccer": 10, "football": 15, "tennis": 8}
5
6 for x in sport:
7     try:
8         print(ppl_play[x])
9     except KeyError:
10        ppl_play[x] = 1
11        print(ppl_play[x])
12

```

```

4
1
10
8
15
1

```

ActiveCode (ac_exceptions_04)

Result	Actual Value	Expected Value	Notes
Pass	[('ba...', 8)]	[('ba...', 8)]	Testing that ppl_play is assigned to correct values.

You passed: 100.0% of the tests

Not yet graded

Provided is a buggy for loop that tries to accumulate some values out of some dictionaries. Insert a try/except so that the code passes. If the key is not there, initialize it in the dictionary and set the value to zero.

Save & Run 5/15/2021, 3:16:57 PM - 2 of 2 Show in CodeLens

```

1
2 di = [{"Puppies": 17, "Kittens": 9, "Birds": 23, "Fish": 90, "Hamsters": 49}, {"Puppies": 0
3 total = 0
4 for diction in di:
5     try:
6         total = total + diction['Puppies']
7     except KeyError:
8         diction['Puppies'] = 0
9
10 print("Total number of puppies:", total)
11
12

```

Total number of puppies: 130

ActiveCode (ac_exceptions_041)

Result	Actual Value	Expected Value	Notes
Pass	4	4	Testing that every dictionary in di has the key 'Puppies'.

You passed: 100.0% of the tests

[Score Me](#)