



EXPLORER

{ } User Settings

percipio48_set_comprehensions.py x

OPEN EDITORS

{ } User Settings C:\Users\pcurtis7\...

percipio48_set_comprehension...

PYTHON

7_01_start

02_Data-Sequence Types

percipio04_int_types.py

percipio05_float_type.py

percipio06_math_functions.py

percipio07_boolean_type.py

percipio08_Strings.py

percipio09_float_type.py

percipio10_bytes_type.py

percipio11_bytearray_type.py

percipio12_list_type.py

percipio13_tuple_type.py

percipio14_slice_type.py

percipio14a_list_copy_boolean...

03_Collections-Mapping-Looping

04_Modules-Functions

05_Classes

06_Working-with-Files

07_Comprehensions

percipio45_list_comprehension...

percipio46_nested_comprehen...

percipio47_zip_function_list_co...

percipio48_set_comprehension...

percipio49_dictionary_compre...

08_Iterables-and-Generators

09_Exceptions

Python Projects 2014

```
1 '''
2 percipio48_set_comprehensions.py
3 Percipio video: Comprehensions; Set Comprehensions
4
5 Demonstrate Set Comprehensions
6 *Set Comprehensions are similar to List Comprehensions with the exception that they generate
   'sets' instead of 'lists' AND the produced values generated are immutable (i.e.: integers,
   strings, tuples).
7
8 * Set notation uses curly bractets{} around the comprehension
9 * The calendar-module is-leap-function returns true when something is a leap year
10 '''
11 nl = '\n'
12 from math import pi # importing the pi function from the math module
13 print(nl, 'Set Comprehension Example 01')
14 pi_limits = {val * pi * 0.1 for val in range(-10, 11)} # creating a set called 'pi_limits'
   where the value * pi * 0.1 for each value in the range from -10 to but not including 11.
   The 0.1 is used because the range() function cannot use float values, it went from -10
   to 10 with each value factored by 0.1 to make the integer smaller.
15 print('pi_limits:', pi_limits) #
16 print('pi in pi_limits:', pi in pi_limits) # to verify pi is in pi_limits, do a truth test
17
18 print(nl, 'Set Comprehension Example 02')
19 from calendar import isleap # from the calendar module the is-leap function is imported
20 leapyears = {year for year in range(2016, 2066) if isleap(year)} # use set Comprehension to
   generate a 'set' of leap years. The year is output for each year within the range, AND
   only if it is a leap year. This code, <if isleap(year)>, is the optional predicate
   clause using 'if' & if the output is True, the output value is included in the set.
21 print('leapyears:', leapyears) # leap years printed out
22 print('2016 in leapyears:', 2016 in leapyears) # True, 2016 is in leapyears
```




EXPLORER

OPEN EDITORS 1 UNSAVED

{ } User Settings C:\Users\pcurtis7\...

percipio48_set_comprehension...

PYTHON

02_Data-Sequence Types

percipio04_int_types.py

percipio05_float_type.py

percipio06_math_functions.py

percipio07_boolean_type.py

percipio08_Strings.py

percipio09_float_type.py

percipio10_bytes_type.py

percipio11_bytearray_type.py

percipio12_list_type.py

percipio13_tuple_type.py

percipio14_slice_type.py

percipio14a_list_copy_boolean...

03_Collections-Mapping-Looping

04_Modules-Functions

05_Classes

06_Working-with-Files

07_Comprehensions

percipio45_list_comprehension...

percipio46_nested_comprehen...

percipio47_zip_function_list_co...

percipio48_set_comprehension...

percipio49_dictionary_compre...

08_Iterables-and-Generators

09_Exceptions

{ } User Settings

percipio48_set_comprehensions.py

```
22
23 print(n1, 'Set Comprehension Example 03')
24 # starting out with existing sets and combining them together
25 basket_a = {'apple', 'orange', 'pear'}
26 basket_b = {'lemon', 'pear', 'peach'}
27 basket = {fruit_a + fruit_b for fruit_a, fruit_b in zip(basket_a, basket_b) if fruit_a !=
    fruit_b} # create a new basket set concatenating fruit_a to fruit_b, where fruit_a &
    fruit_b are extracted from zipping basket_a & basket_b together. The concatenation or
    output only happens if fruit_a is not equal to fruit_b
28 print('basket:', basket) # Combines the two sets, basket_a & basket_b, in a random order.
    If pear & pear are matched, it will not print
29
30 RESULTS:
31 Set Comprehension Example 01
32 pi_limits: {-0.9424777960769379, -0.6283185307179586, -0.3141592653589793, 0.0,
    0.3141592653589793, 0.6283185307179586, 0.9424777960769379, 1.2566370614359172,
    1.5707963267948966, 1.8849555921538759, 2.199114857512855, 2.5132741228718345,
    2.827433388230814, 3.141592653589793, -1.8849555921538759, -2.5132741228718345,
    -2.199114857512855, -1.5707963267948966, -3.141592653589793, -2.827433388230814,
    -1.2566370614359172}
33 pi in pi_limits: True
34
35 Set Comprehension Example 02
36 leapyears: {2016, 2048, 2020, 2052, 2024, 2056, 2028, 2060, 2032, 2064, 2036, 2040, 2044}
37 2016 in leapyears: True
38
39 Set Comprehension Example 03
40 basket: {'pearpeach', 'applepear', 'orangelemon'}
41
```