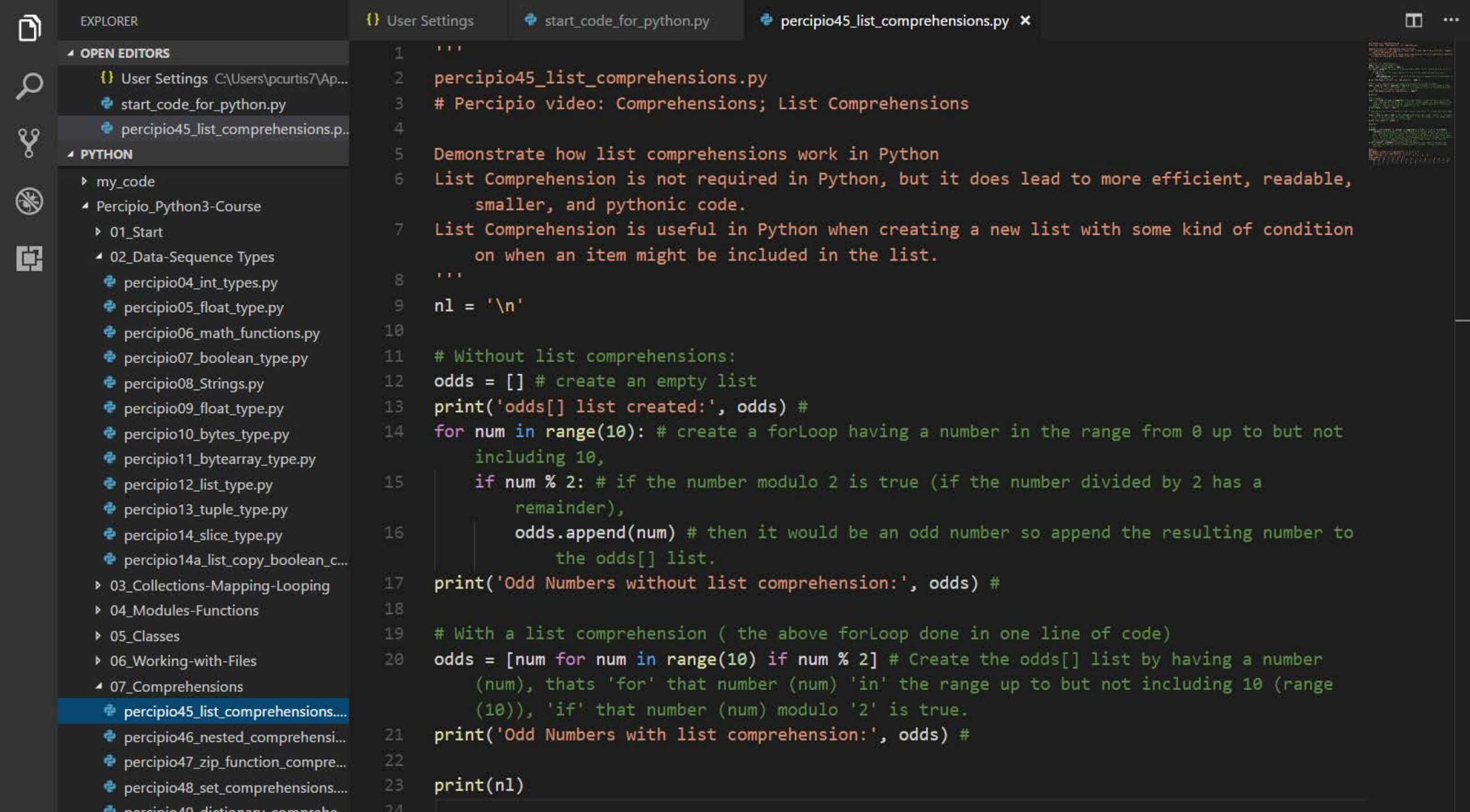
EXPLORER
OPEN EDITORS
User Settings C:\Users\pcurtis7\Ap...
start_code_for_python.py
percipio45_list_comprehensions.p...
PYTHON
my_code
Percipio_Python3-Course
01_Start
02_Data-Sequence Types
percipio04_int_types.py
percipio05_float_type.py
percipio06_math_functions.py
percipio07_boolean_type.py
percipio08_Strings.py
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_c...
03_Collections-Mapping-Looping
04_Modules-Functions
05_Classes
06_Working-with-Files
07_Comprehensions
percipio45_list_comprehensions....
percipio46_nested_comprehensi...
percipio47_zip_function_compre...
percipio48_set_comprehensions....
User Settings
start_code_for_python.py
percipio45_list_comprehensions.py
'''
percipio45_list_comprehensions.py
Percipio video: Comprehensions; List Comprehensions
Demonstrate how list comprehensions work in Python
List Comprehension is not required in Python, but it does lead to more efficient, readable, smaller, and pythonic code.
List Comprehension is useful in Python when creating a new list with some kind of condition on when an item might be included in the list.
'''
nl = '\n'
# Without list comprehensions:
odds = [] # create an empty list
print('odds[] list created:', odds) #
for num in range(10): # create a forLoop having a number in the range from 0 up to but not including 10,
if num % 2: # if the number modulo 2 is true (if the number divided by 2 has a remainder),
odds.append(num) # then it would be an odd number so append the resulting number to the odds[] list.
print('Odd Numbers without list comprehension:', odds) #
# With a list comprehension (the above forLoop done in one line of code)
odds = [num for num in range(10) if num % 2] # Create the odds[] list by having a number (num), thats 'for' that number (num) 'in' the range up to but not including 10 (range (10)), 'if' that number (num) modulo '2' is true.
print('Odd Numbers with list comprehension:', odds) #
print(nl)

EXPLORER

{} User Settings | start_code_for_python.py | percipio45_list_comprehensions.py ●

OPEN EDITORS 1 UNSAVED
- {} User Settings C:\Users\pcurtis7\Ap...
- start_code_for_python.py
- ● percipio45_list_comprehensions.p...

PYTHON
- my_code
- Percipio_Python3-Course
  - 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_c...
  - 03_Collections-Mapping-Looping
  - 04_Modules-Functions
  - 05_Classes
  - 06_Working-with-Files
  - 07_Comprehensions
    - percipio45_list_comprehensions....
    - percipio46_nested_comprehensi...
    - percipio47_zip_function_compre...
    - percipio48_set_comprehensions....
    - percipio49_dictionary_comprehe...
  - 08_Iterables-and-Generators
  - 09_Exceptions
- Python Projects_2014
- CMD_Python_Set-Path.txt
- Python_Clear-Window-Command.txt
- python_exercises_00.py
- python_exercises_01.py
- Python_Tutorial_Running-Scripts.docx
- Python_Tutorials.md
- start_code_for_python.py

```
23  print(nl)
24
25  # More Examples:
26  nums = [str(num) for num in range(10)] # To create number strings, set up a list called
        'nums' that applies the string (str) function, to each number for those numbers in the
        range up to but not including 10.  The result is all numbers have the string function
        applied to them and then they are added to the list
27  print('Number strings:', nums) #
28
29  # Take the previous list of number strings (nums[]), and filter it to have only even number
        strings
30  evens = [num for num in nums if not int(num) % 2] # have each number (num-#2) in the nums[]
        list that would include that number (num-#1), if not the integer of that number modulo
        2. If it's an even number, it is included.
31  print('Even numbers:', evens) #
32
33  print(nl)
34
35  # 03:05
36  alphabet = [chr(ordinal) for ordinal in range(ord('A'), ord('z') + 1) if chr(ordinal)
        .isalpha()] # an alphabet[] list thats generated by using a character function, or
        'chr', applied to an ordinal, where the ordinal is a number in the range starting at
        where the ordinal value of capital 'A' is, which is 65, going up to the ordinal value
        of 'z', which is 122+1. So will go from the ordinal value of 'A' through 'z' because of
        the +1.  It will include the character with that ordinal value, if the character
        function applied to ordinal is an alpha character.
37  print('Alphabet', alphabet) # This will exclude some punction symbols in between the
        uppercase and lowercase letters ending up with an Alphabet list having only alpha
        characters.
38  '''
39  RESULTS:
40  Odd Numbers without list comprehension: [1, 3, 5, 7, 9]
41  Odd Numbers with list comprehension: [1, 3, 5, 7, 9]
42  Number strings: ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9']
43  Even numbers: ['0', '2', '4', '6', '8']
44  Alphabet ['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', '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']
45
46  '''
```