

Course Name	ITD 2313 – Script Programming
Instructor	Andy Tripp
Student Name	Timothy Obinda
Due date	10/19/2025
Grade	Put grade earned here
Grading Comments	Put instructor comments here

INSTRUCTIONS FOR THE EXERCISE

You should always read the instructions in full. It is best to do a full read through before starting the assignment. Each screen shot needs to be appropriately labeled.

In these instruction, you are given information to execute specific examples in the text. You are given the Section, Subsection and a page number to identify a set of steps. On each page listed, there will be 1 or more numbered tasks to perform. These numbered tasks will be what you are to type in, execute and then grab the screen shots of. Those screen shots will the go into your submission document.

Some advice, copy this instruction set into your submission document and then put the screen shots under each numbered task. Each individual book page in the instructions should be in a different screen shot. For any single book page, you may have all the numbered tasks on that page be in a single screen shot.

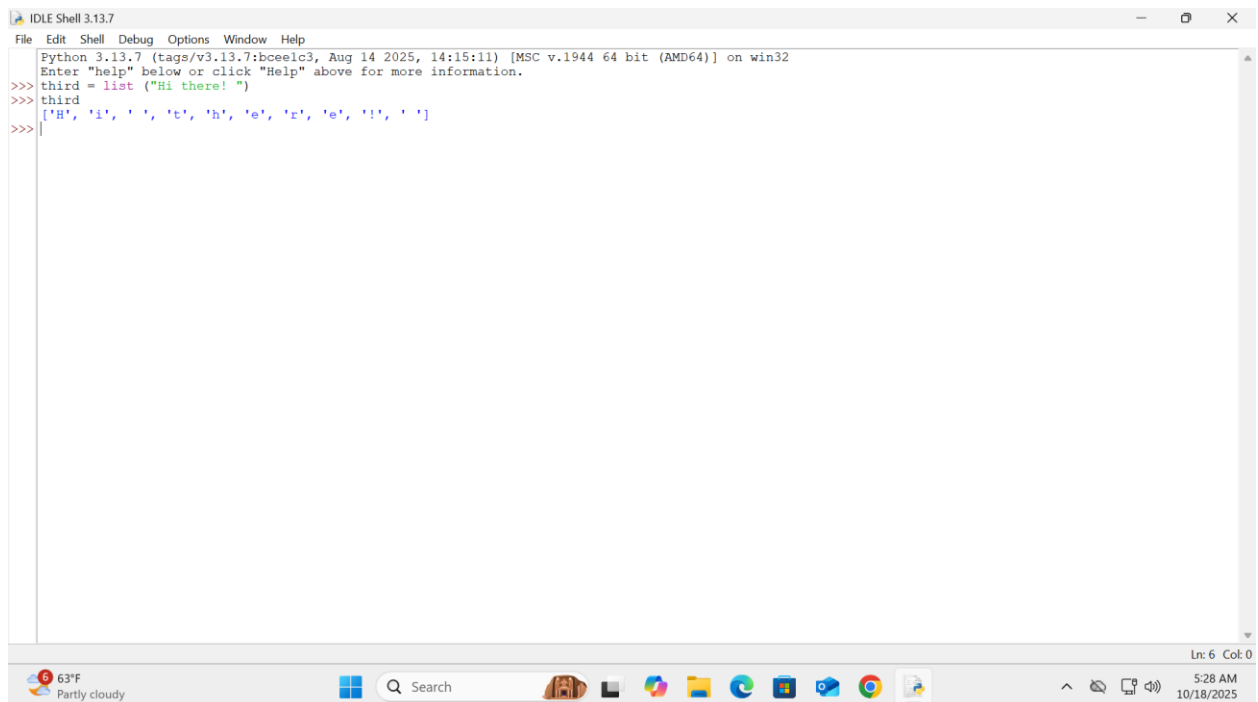
Lists

List Literals and Basic Operations

Page 116 -117

1. There are 8 small sets of code example blocks for this section. The first is about 3/4 way down and starts with Import math.

2. The second code example block is at the bottom and spills over to the top of the next page (117) and starts with `first=`.
3. The third code example block is just a few lines later and starts with `third=`.



The screenshot shows a Windows desktop with the IDLE Shell 3.13.7 application open. The window title is "IDLE Shell 3.13.7". The menu bar includes "File", "Edit", "Shell", "Debug", "Options", "Window", and "Help". The shell area displays the following text:

```
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>> third = list ("Hi there! ")
>>> third
['H', 'i', ' ', 't', 'h', 'e', 'r', 'e', '!', ' ']
```

The Windows taskbar is visible at the bottom, showing the weather (63°F, Partly cloudy), a search bar, and several application icons. The system clock in the bottom right corner indicates 5:28 AM on 10/18/2025.

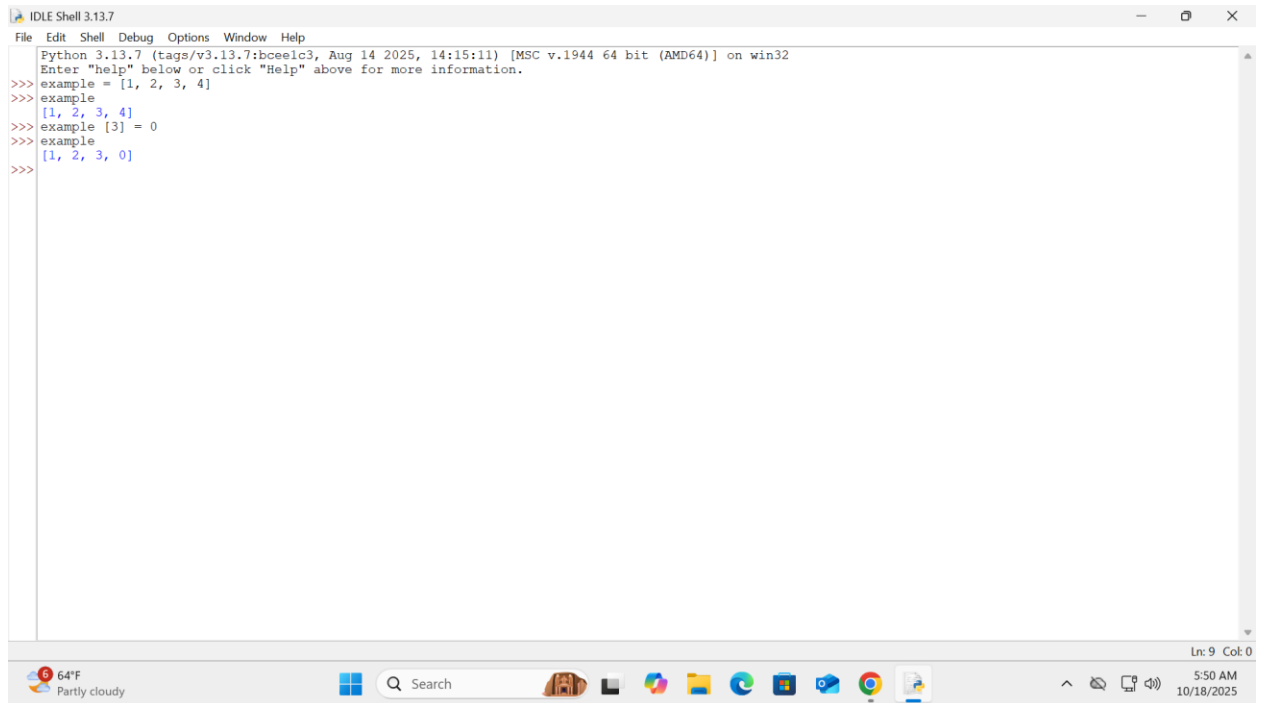
4. The fourth code example block starts with the `len` function.
5. The fifth code example block starts with `first +`.
6. The sixth code example block will be the one that starts `print`.
7. The seventh code example block is the one that covers the `for` loop.
8. The last code example block on this page deals with the `in` operator.

```
IDLE Shell 3.13.7
File Edit Shell Debug Options Window Help
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>> first = [1, 2, 3, 4]
>>> second = list(range(1, 5))
>>> first
[1, 2, 3, 4]
>>> second
[1, 2, 3, 4]
>>> len(first)
4
>>> first[0]
1
>>> first[2:4]
[3, 4]
>>> first + [5, 6]
[1, 2, 3, 4, 5, 6]
>>> first == second
True
>>> print("1234")
1234
>>> print([1, 2, 3, 4])
[1, 2, 3, 4]
>>> for number in [1, 2, 3, 4]:
...     print(number, end = " ")
...
...
1 2 3 4
>>> 3 in [1, 2, 3]
True
>>> 0 in [1, 2, 3]
False
>>>
```

Replacing an Element in a List

Page 118

1. There are three code example blocks for this section, the first being about 3/4 the way down on the page.

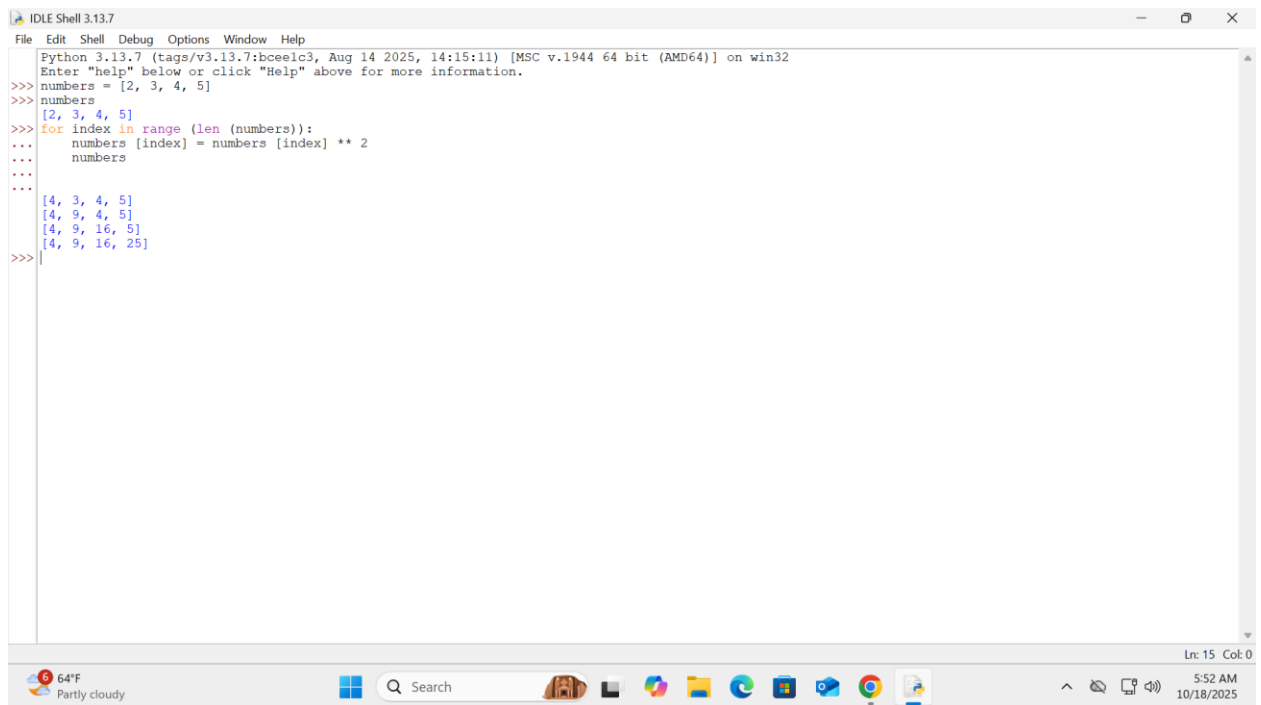


The screenshot shows the IDLE Shell 3.13.7 window. The menu bar includes File, Edit, Shell, Debug, Options, Window, and Help. The shell text area contains the following Python code and its output:

```
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>> example = [1, 2, 3, 4]
>>> example
[1, 2, 3, 4]
>>> example[3] = 0
>>> example
[1, 2, 3, 0]
>>>
```

The Windows taskbar at the bottom shows the weather as 64°F, Partly cloudy, and the system clock as 5:50 AM on 10/18/2025.

2. The second code example block is at the bottom of the page.



The screenshot shows the IDLE Shell 3.13.7 window. The menu bar includes File, Edit, Shell, Debug, Options, Window, and Help. The shell text area contains the following Python code and its output:

```
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>> numbers = [2, 3, 4, 5]
>>> numbers
[2, 3, 4, 5]
>>> for index in range(len(numbers)):
...     numbers[index] = numbers[index] ** 2
...     numbers
...
[4, 3, 4, 5]
[4, 9, 4, 5]
[4, 9, 16, 5]
[4, 9, 16, 25]
>>>
```

The Windows taskbar at the bottom shows the weather as 64°F, Partly cloudy, and the system clock as 5:52 AM on 10/18/2025.

3. The last code example block on the page is at the top of page 119.

```
IDLE Shell 3.13.7
File Edit Shell Debug Options Window Help
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>> sentence = " This example has five words. "
>>> words = sentence.split ()
>>> words
['This', 'example', 'has', 'five', 'words.']
>>> for index in range (len(words)):
...     words [index] = words [index].upper ()
...
...
>>>
['THIS', 'example', 'has', 'five', 'words.']
['THIS', 'EXAMPLE', 'has', 'five', 'words.']
['THIS', 'EXAMPLE', 'HAS', 'five', 'words.']
['THIS', 'EXAMPLE', 'HAS', 'FIVE', 'words.']
['THIS', 'EXAMPLE', 'HAS', 'FIVE', 'WORDS.'].
>>>
```

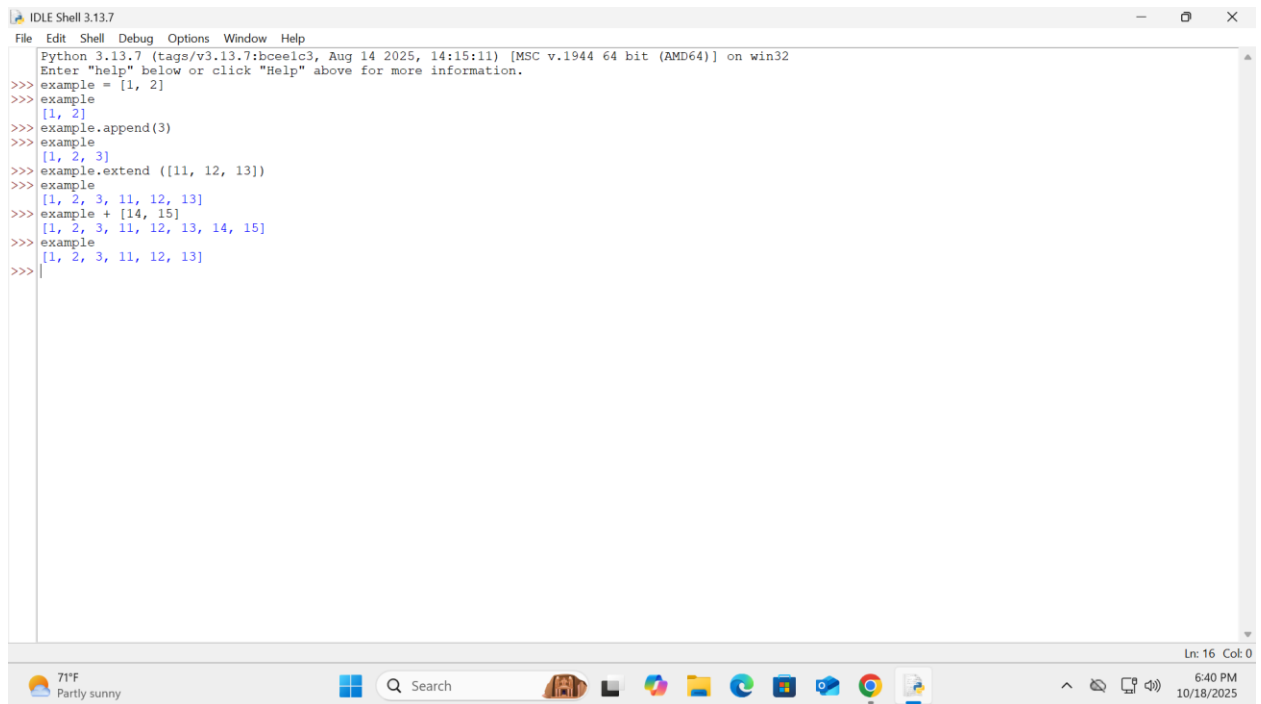
List Methods for Inserting and Removing Elements

Page 119 - 120

1. There are three code example blocks on this page, the first is at the bottom of the page.

```
IDLE Shell 3.13.7
File Edit Shell Debug Options Window Help
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>> example = [1, 2]
>>> example
[1, 2]
>>> example.insert(1, 10)
>>> example
[1, 10, 2]
>>> example.insert (3, 25)
>>> example
[1, 10, 2, 25]
>>>
```

2. The second code example block is on the top of page 120.

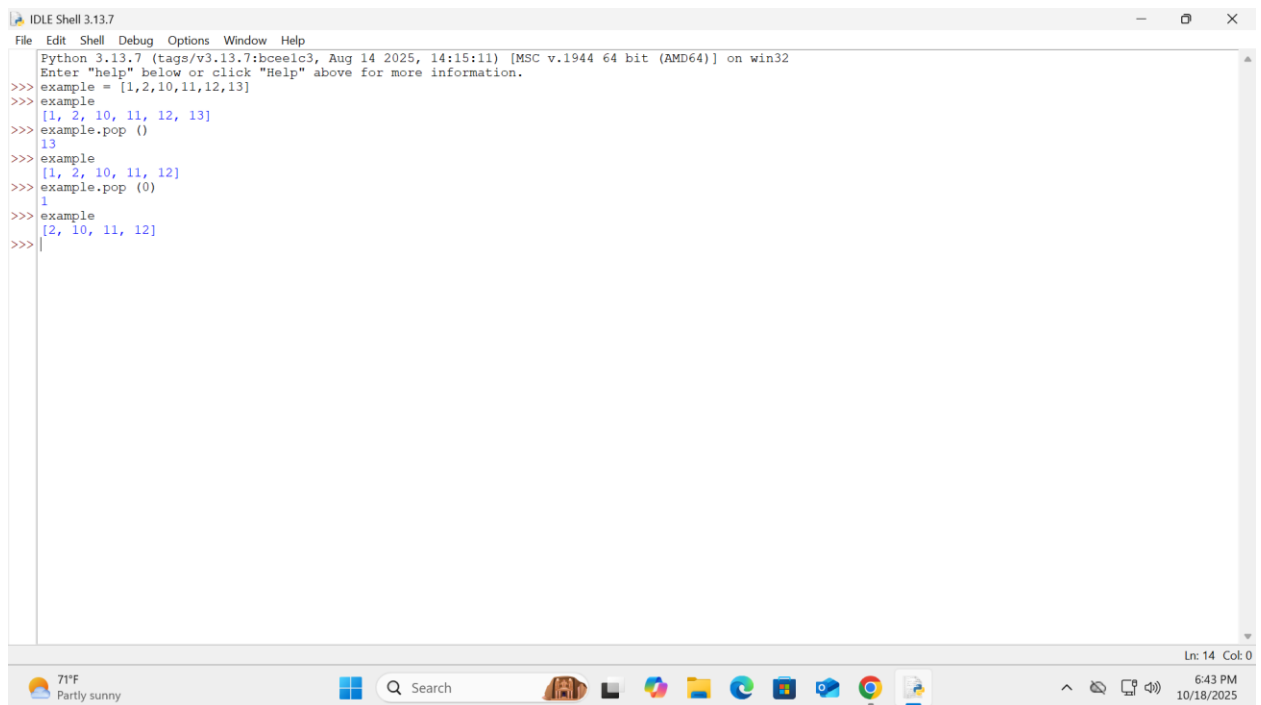


```
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>> example = [1, 2]
>>> example
[1, 2]
>>> example.append(3)
>>> example
[1, 2, 3]
>>> example.extend([11, 12, 13])
>>> example
[1, 2, 3, 11, 12, 13]
>>> example + [14, 15]
[1, 2, 3, 11, 12, 13, 14, 15]
>>> example
[1, 2, 3, 11, 12, 13]
>>>
```

Ln: 16 Col: 0

71°F Partly sunny 6:40 PM 10/18/2025

3. The last is about halfway down the page.



```
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>> example = [1, 2, 10, 11, 12, 13]
>>> example
[1, 2, 10, 11, 12, 13]
>>> example.pop()
13
>>> example
[1, 2, 10, 11, 12]
>>> example.pop(0)
1
>>> example
[2, 10, 11, 12]
>>>
```

Ln: 14 Col: 0

71°F Partly sunny 6:43 PM 10/18/2025

Searching a list

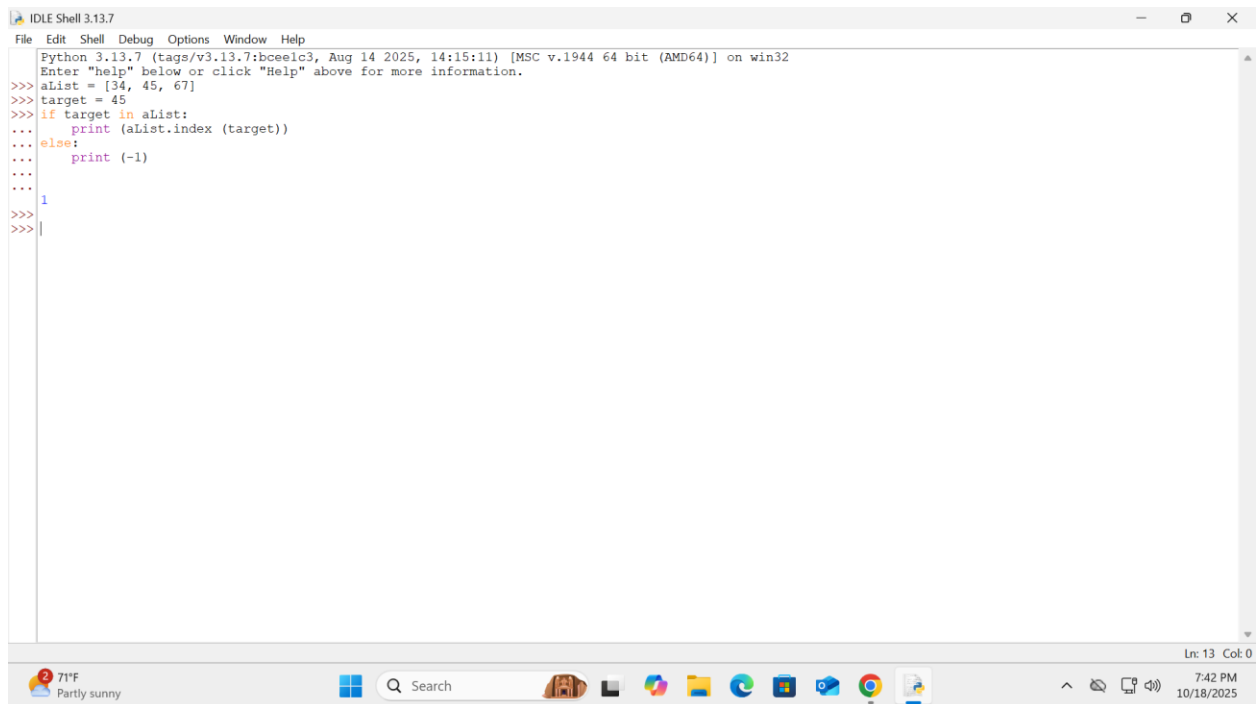
Page 121

1. The code example block is at the top of the page. It does not show output. You will need to show the output of that search code.

Sorting a list

Page 121

1. The first code example block is about half way down the page.

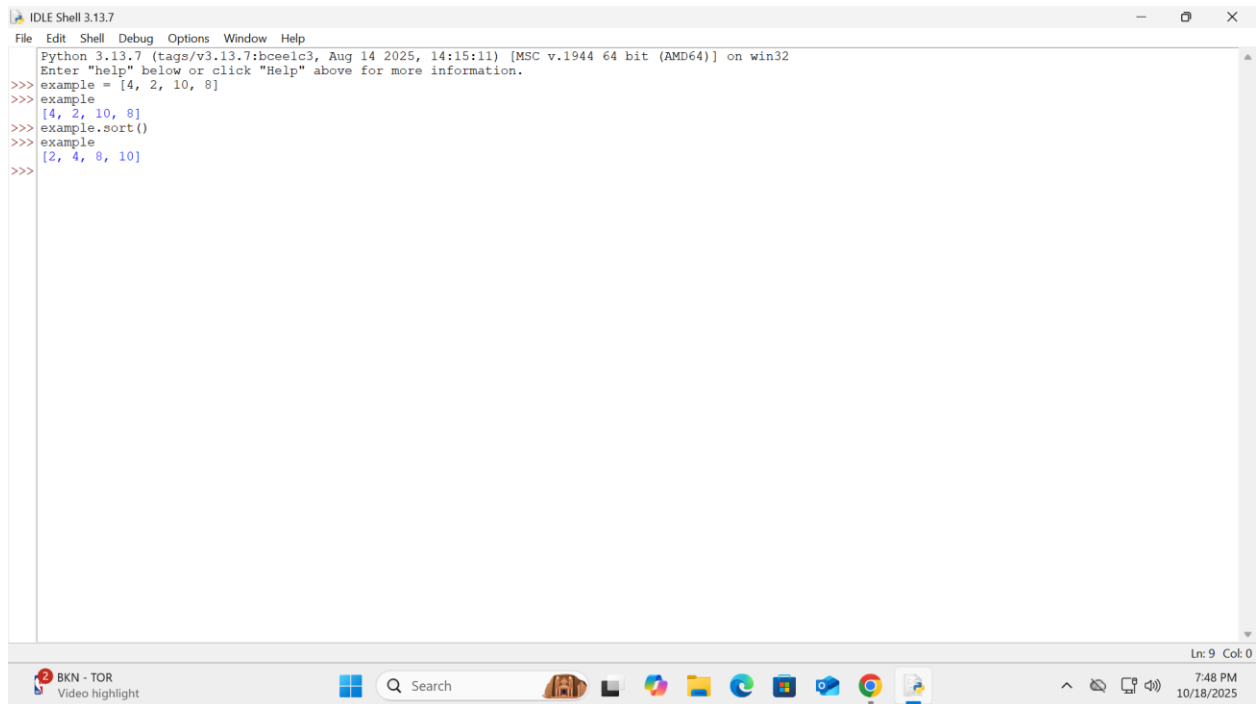


The screenshot shows a Python IDLE Shell window with the following code and output:

```
Python 3.13.7 (tags/v3.13.7:bce1c3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>> alist = [34, 45, 67]
>>> target = 45
>>> if target in alist:
...     print (alist.index (target))
... else:
...     print (-1)
...
... 1
>>>
>>> |
```

The window title is "IDLE Shell 3.13.7". The bottom status bar shows "Ln: 13 Col: 0", "71°F Partly sunny", and the date "10/18/2025".

2. The last code example block is on the bottom of the page but it is included on the next set of instructions as it ends on the next page. So just the one screen shot here on this page.



The screenshot shows a Windows desktop with an IDLE Shell window open. The window title is "IDLE Shell 3.13.7". The menu bar includes File, Edit, Shell, Debug, Options, Window, and Help. The shell displays the following text:

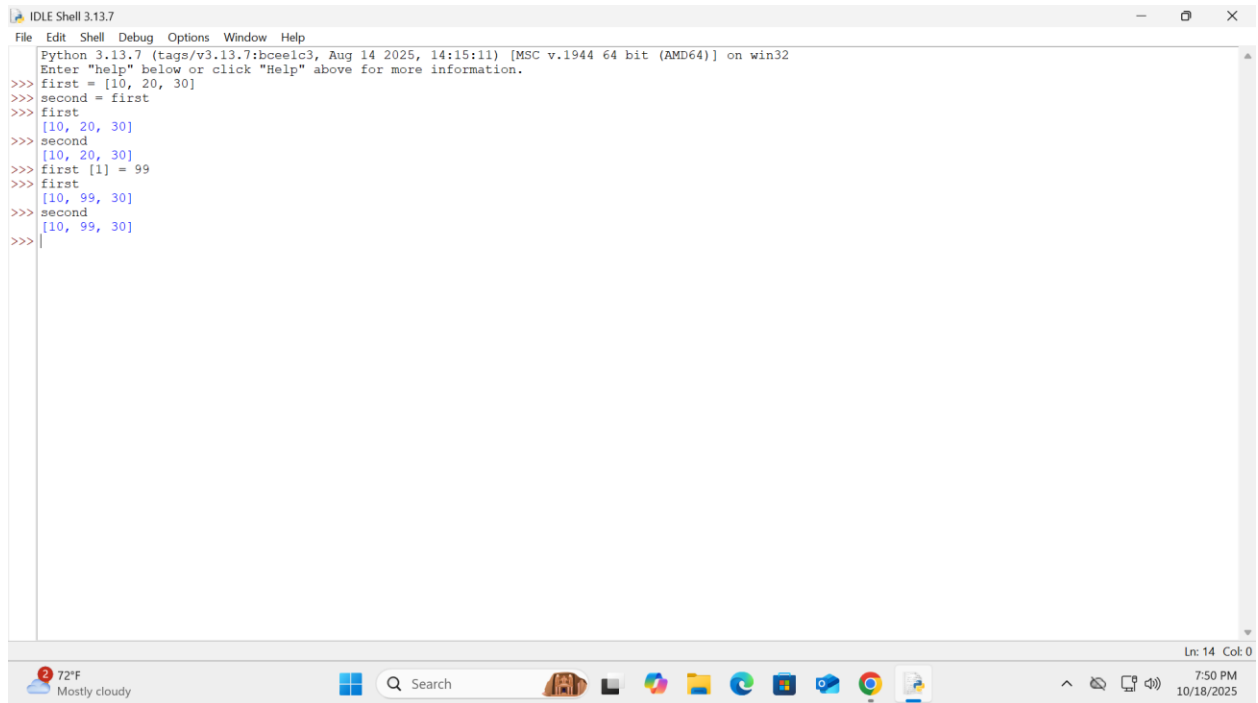
```
Python 3.13.7 (tags/v3.13.7:bce1c3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>> example = [4, 2, 10, 8]
>>> example
[4, 2, 10, 8]
>>> example.sort()
>>> example
[2, 4, 8, 10]
>>>
```

The Windows taskbar is visible at the bottom, showing the Start button, a search bar, and several application icons. The system tray on the right indicates the time is 7:48 PM on 10/18/2025.

Aliasing and Side effects

Page 122

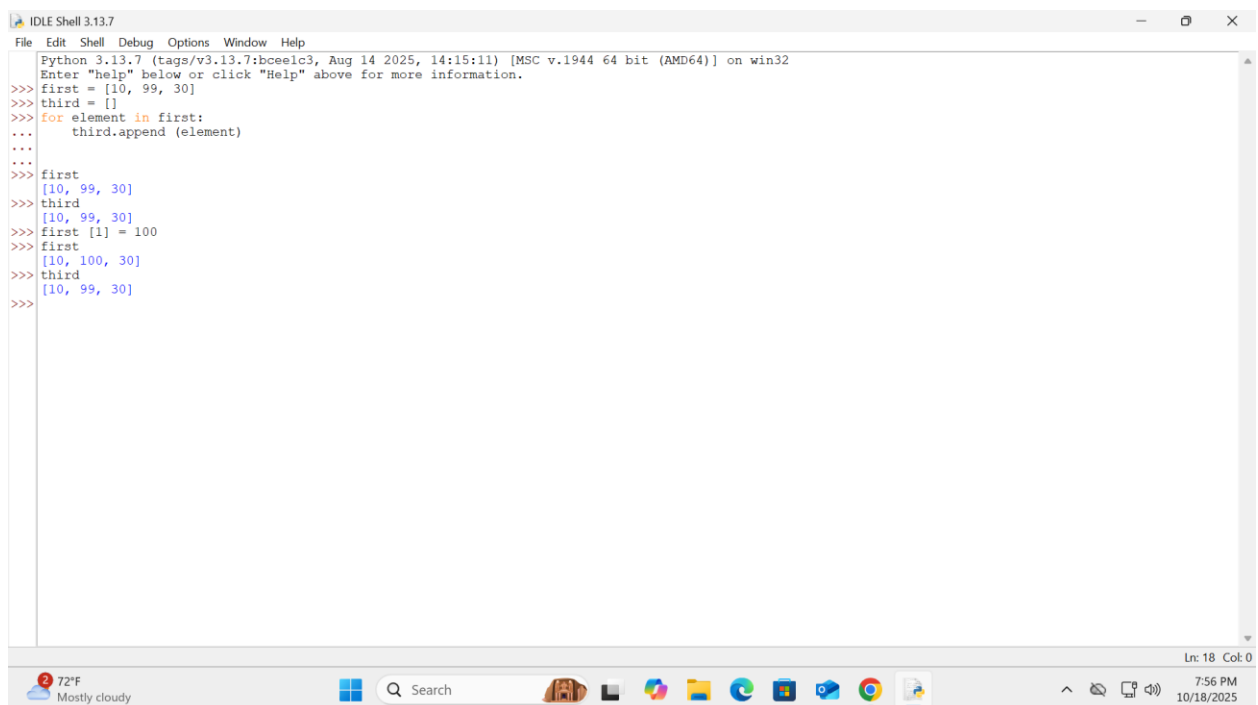
1. The code example block that is at the top of this page. Put the screenshot of the whole set here under this set of instructions.



```
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>> first = [10, 20, 30]
>>> second = first
>>> first
[10, 20, 30]
>>> second
[10, 20, 30]
>>> first[1] = 99
>>> first
[10, 99, 30]
>>> second
[10, 99, 30]
>>>
```

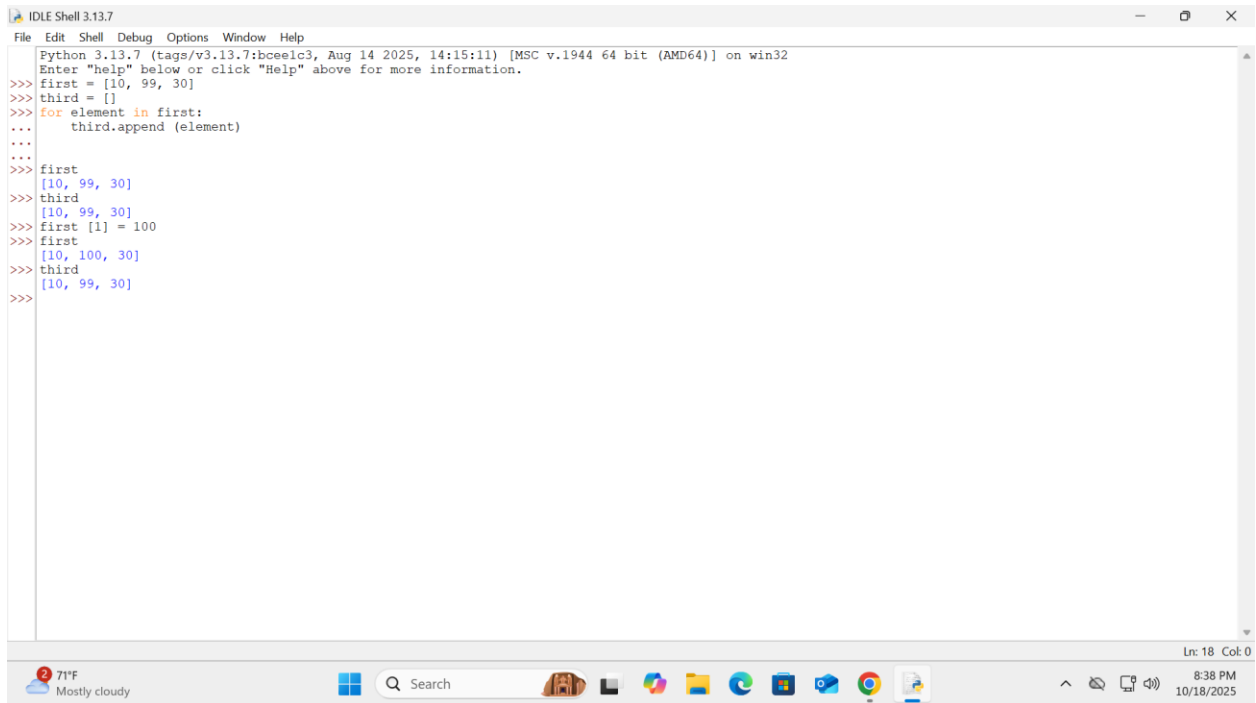
The screenshot shows the IDLE Shell 3.13.7 window. The code defines a list 'first' with values [10, 20, 30] and assigns it to 'second'. Then, the second element of 'first' is changed to 99. The output shows that both 'first' and 'second' now contain [10, 99, 30], demonstrating list aliasing. The Windows taskbar at the bottom shows a temperature of 72°F, 'Mostly cloudy' weather, and the time 7:50 PM on 10/18/2025.

2. About 3/4 way down is the second and final code example block on this page.



```
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>> first = [10, 99, 30]
>>> third = []
>>> for element in first:
...     third.append(element)
...
>>> first
[10, 99, 30]
>>> third
[10, 99, 30]
>>> first[1] = 100
>>> first
[10, 100, 30]
>>> third
[10, 99, 30]
>>>
```

The screenshot shows the IDLE Shell 3.13.7 window. The code creates a list 'first' with values [10, 99, 30] and an empty list 'third'. It then uses a for loop to append each element of 'first' to 'third'. After the loop, 'first' is modified so its second element is 100. The output shows that 'first' is now [10, 100, 30] and 'third' remains [10, 99, 30], demonstrating that 'third' is a new list containing copies of the elements from 'first'. The Windows taskbar at the bottom shows a temperature of 72°F, 'Mostly cloudy' weather, and the time 7:56 PM on 10/18/2025.



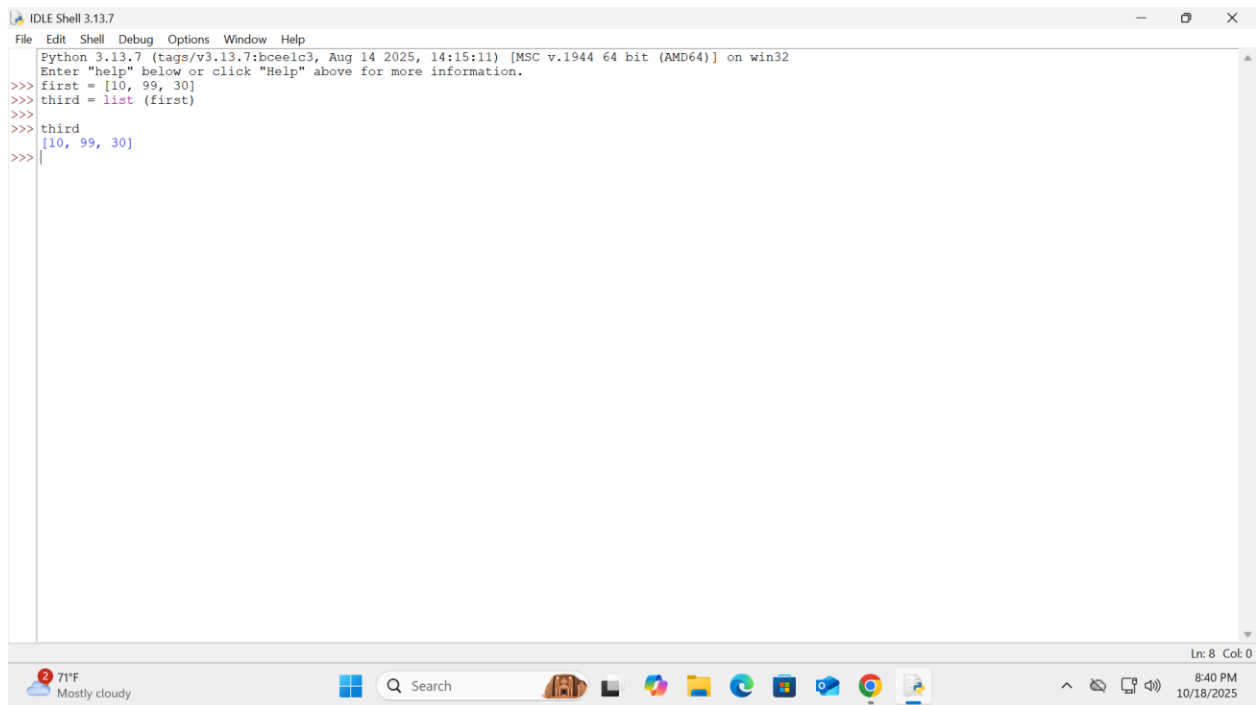
The screenshot shows the IDLE Python Shell interface. The menu bar includes File, Edit, Shell, Debug, Options, Window, and Help. The shell window contains the following Python code:

```
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>> first = [10, 99, 30]
>>> third = []
>>> for element in first:
...     third.append(element)
...
>>> first
[10, 99, 30]
>>> third
[10, 99, 30]
>>> first[1] = 100
>>> first
[10, 100, 30]
>>> third
[10, 99, 30]
>>>
```

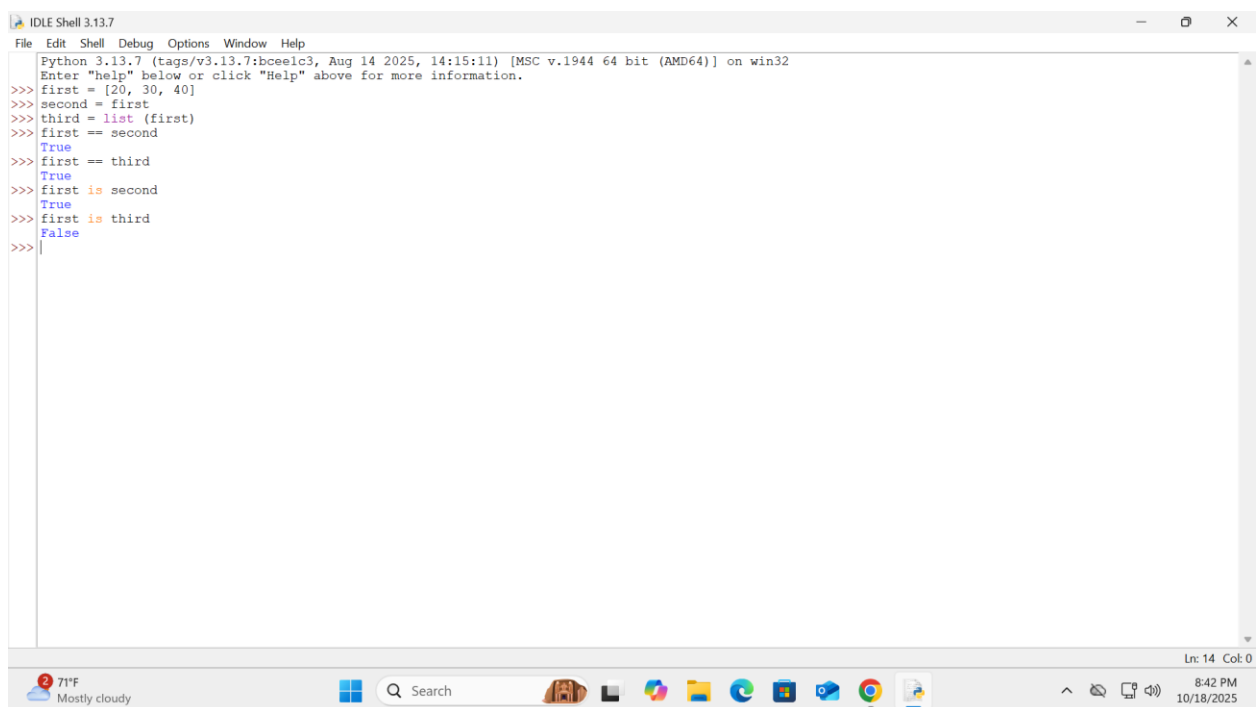
The Windows taskbar at the bottom shows the system clock as 8:38 PM on 10/18/2025, along with weather information (71°F, Mostly cloudy) and various application icons.

Page 123

1. There is a small code example block of a single line at the top.
There is not output given but you will need to execute and catch the output in your screen shot.



```
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>> first = [10, 99, 30]
>>> third = list(first)
>>>
>>> third
[10, 99, 30]
>>>
```



```
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>> first = [20, 30, 40]
>>> second = first
>>> third = list(first)
>>> first == second
True
>>> first == third
True
>>> first is second
True
>>> first is third
False
>>>
```

Page 124

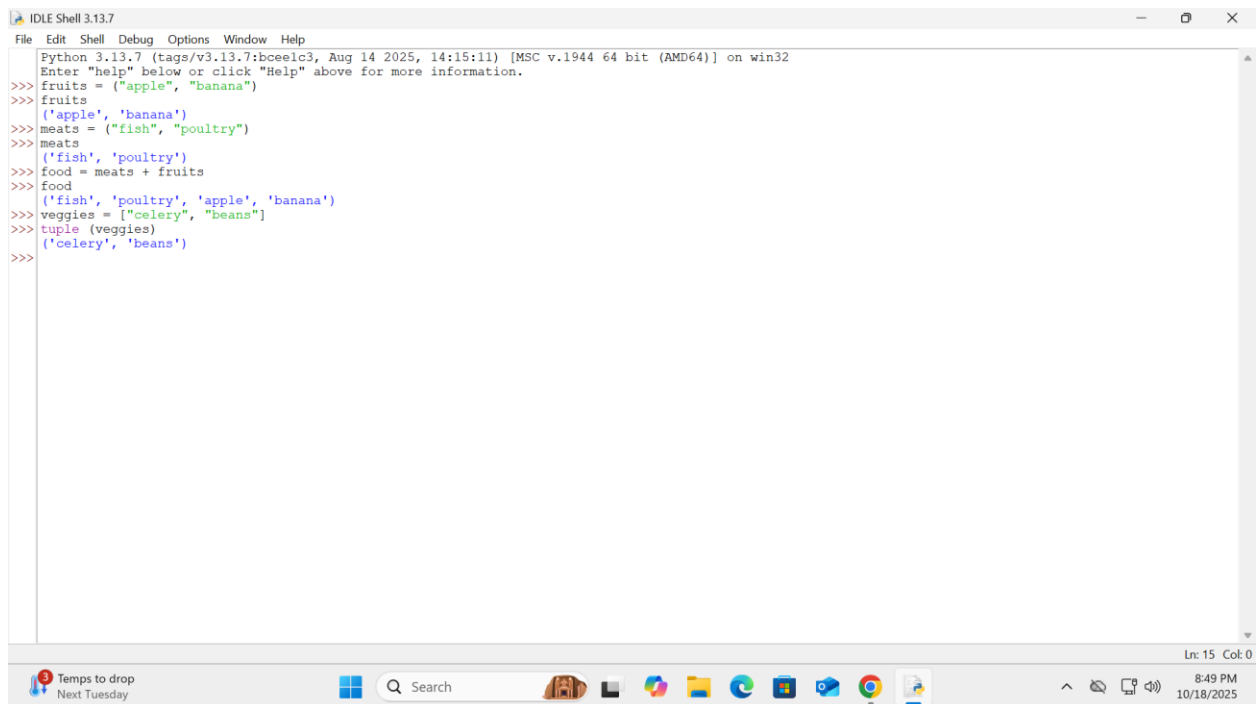
1. The first code example block is actually a script that is available in the download for this chapter. Load it in and run it to get the screen shot.

Download could not open in python

Tuples

Page 124

1. The first code example block starts at the bottom and spills over to the top of the next page. The screenshot for the entire code block should be on this page.

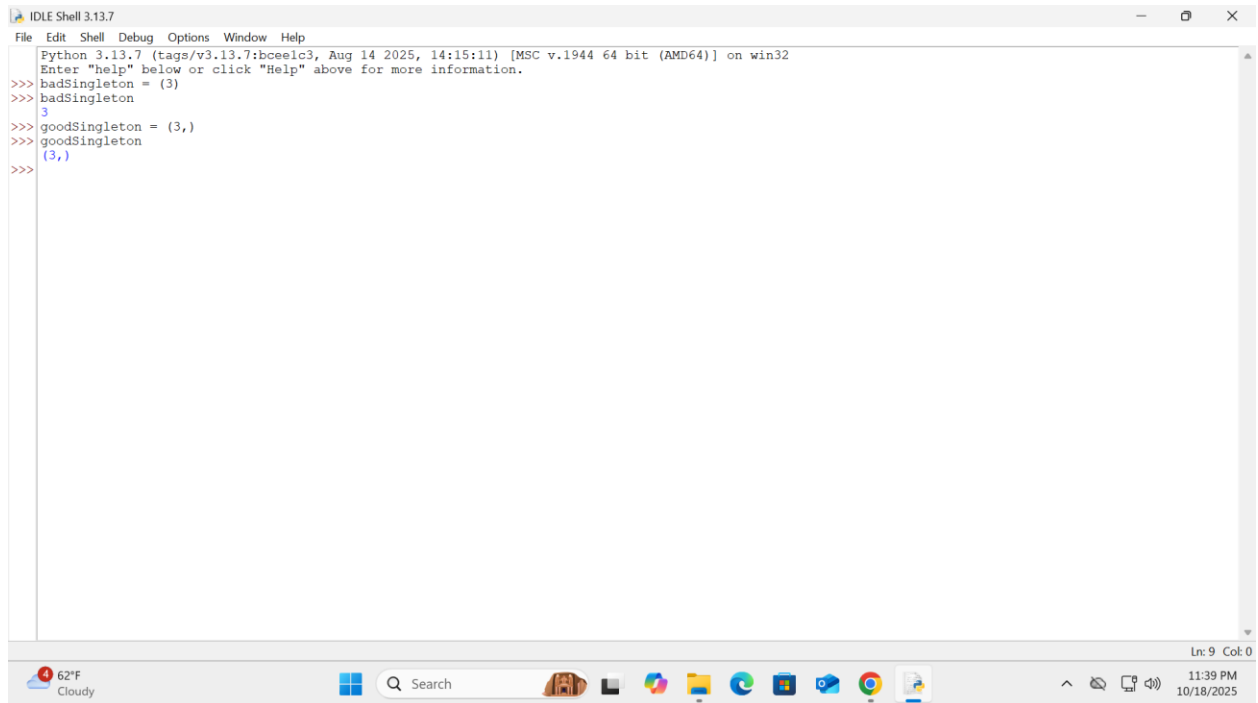


The screenshot shows an IDLE Shell window titled "IDLE Shell 3.13.7". The window contains the following Python code:

```
Python 3.13.7 (tags/v3.13.7:bce51c3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>> fruits = ("apple", "banana")
>>> fruits
('apple', 'banana')
>>> meats = ("fish", "poultry")
>>> meats
('fish', 'poultry')
>>> food = meats + fruits
>>> food
('fish', 'poultry', 'apple', 'banana')
>>> veggies = ["celery", "beans"]
>>> tuple(veggies)
('celery', 'beans')
>>>
```

The window's status bar at the bottom right shows "Ln: 15 Col: 0". The Windows taskbar is visible at the bottom of the screen, showing the Start button, a search bar, and several application icons. A notification in the bottom left corner says "Temps to drop Next Tuesday".

2. The second code example block is about 1/3 of the way and just before the exercises portion on the page.



The screenshot shows an IDLE Shell window titled "IDLE Shell 3.13.7". The menu bar includes File, Edit, Shell, Debug, Options, Window, and Help. The shell displays the following text:

```
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>> badSingleton = (3)
>>> badSingleton
3
>>> goodSingleton = (3,)
>>> goodSingleton
(3,)
```

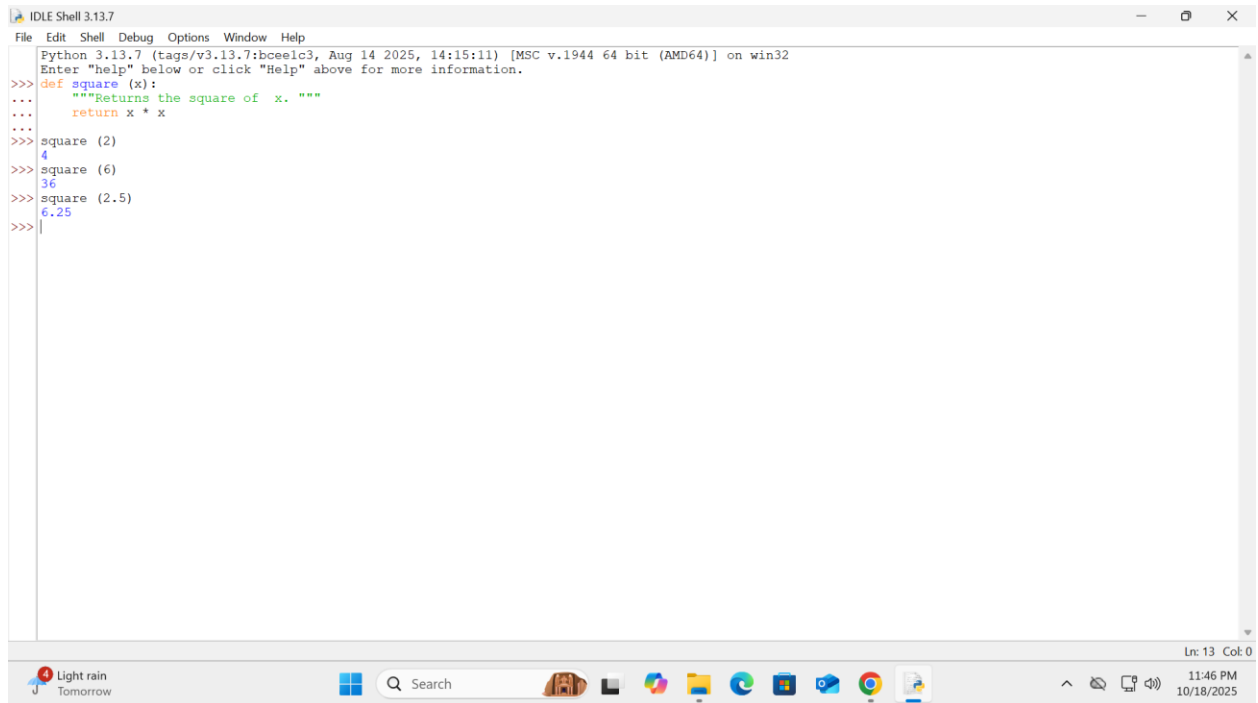
The status bar at the bottom shows "Ln: 9 Col: 0". The Windows taskbar is visible at the bottom, displaying the date and time as 11:39 PM on 10/18/2025, along with weather information (62°F, Cloudy) and various application icons.

Defining Simple Functions

The Syntax of Simple Function Definitions

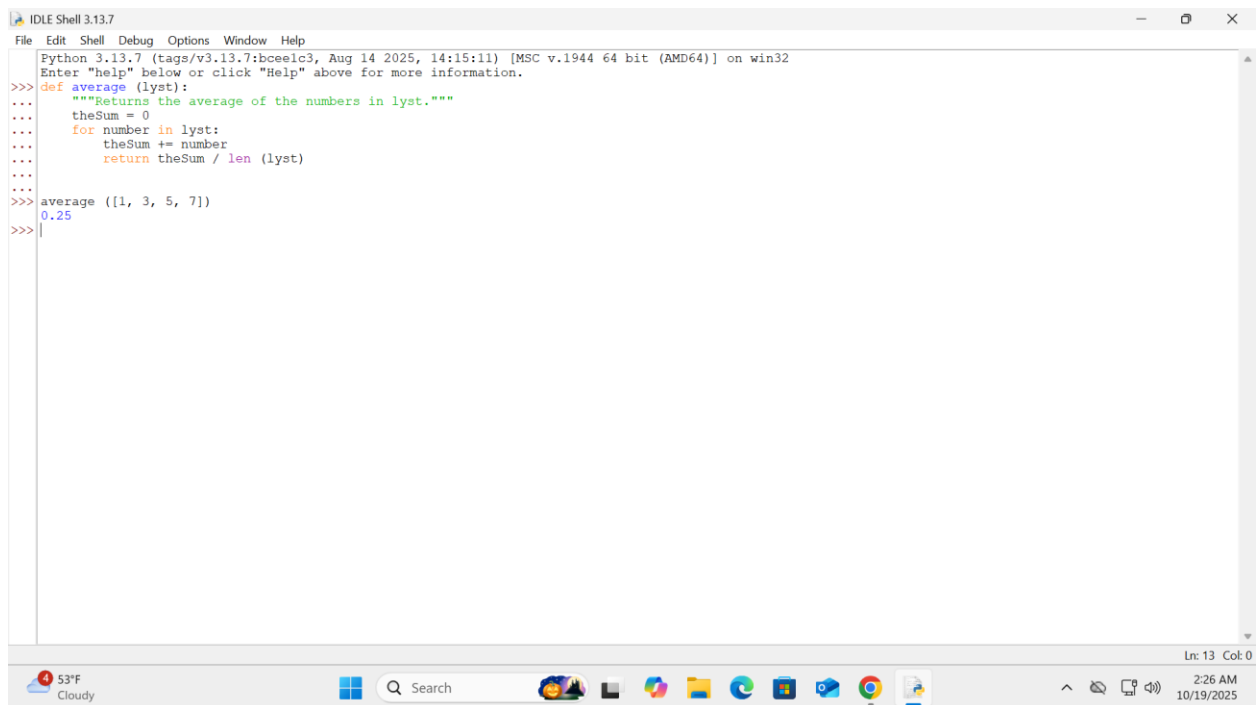
Page 126

1. You will need to do the small code example block about 1/2 of the way page before doing the few lines above it. This last code example block on the page defines the function that is used in those previous lines.



```
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>> def square(x):
...     """Returns the square of x."""
...     return x * x
...
>>> square(2)
4
>>> square(6)
36
>>> square(2.5)
6.25
>>>
```

2. The code example block in at the bottom of the page and spills over to the top of the next page is similar to the one above. In that the function needs to be typed in first and then the line above it so it will execute correctly.

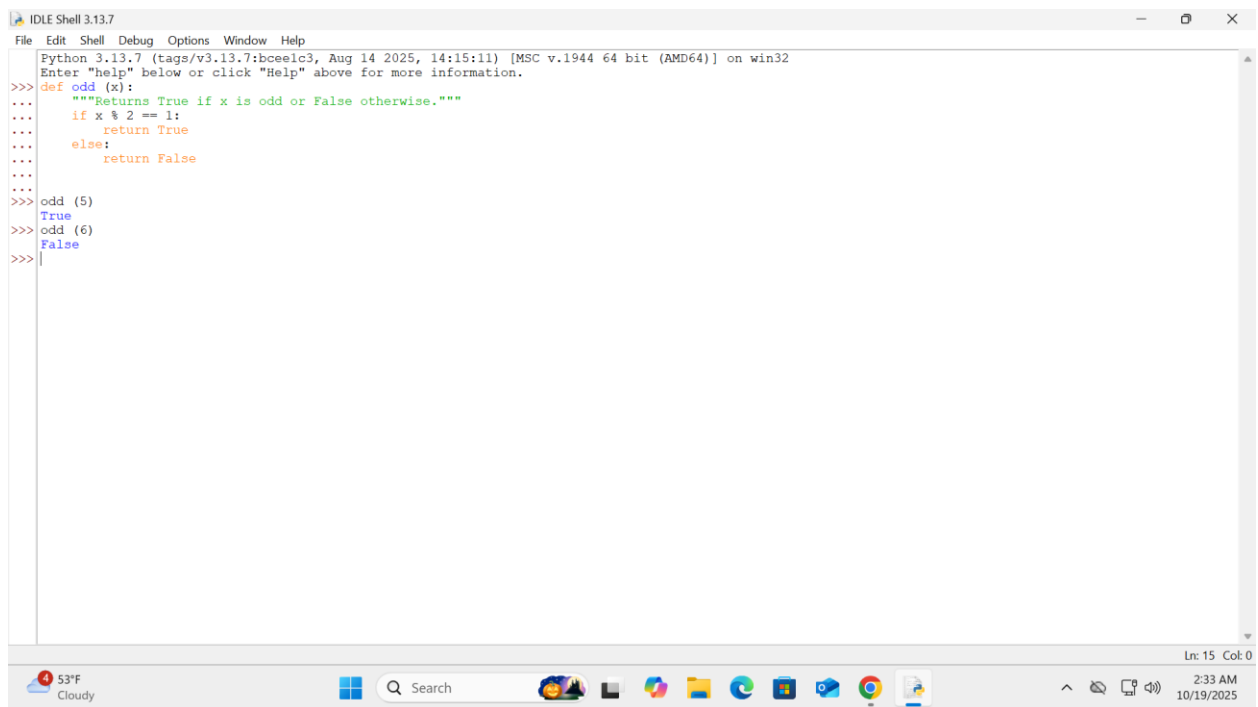


```
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>> def average(lyst):
...     """Returns the average of the numbers in lyst."""
...     theSum = 0
...     for number in lyst:
...         theSum += number
...     return theSum / len(lyst)
...
>>> average([1, 3, 5, 7])
0.25
>>>
```

Boolean Function

Page 127

1. The code example block in the middle of this page also has that function example after the lines that use it. So you will need to type in the function items first and then do the examples using the function.

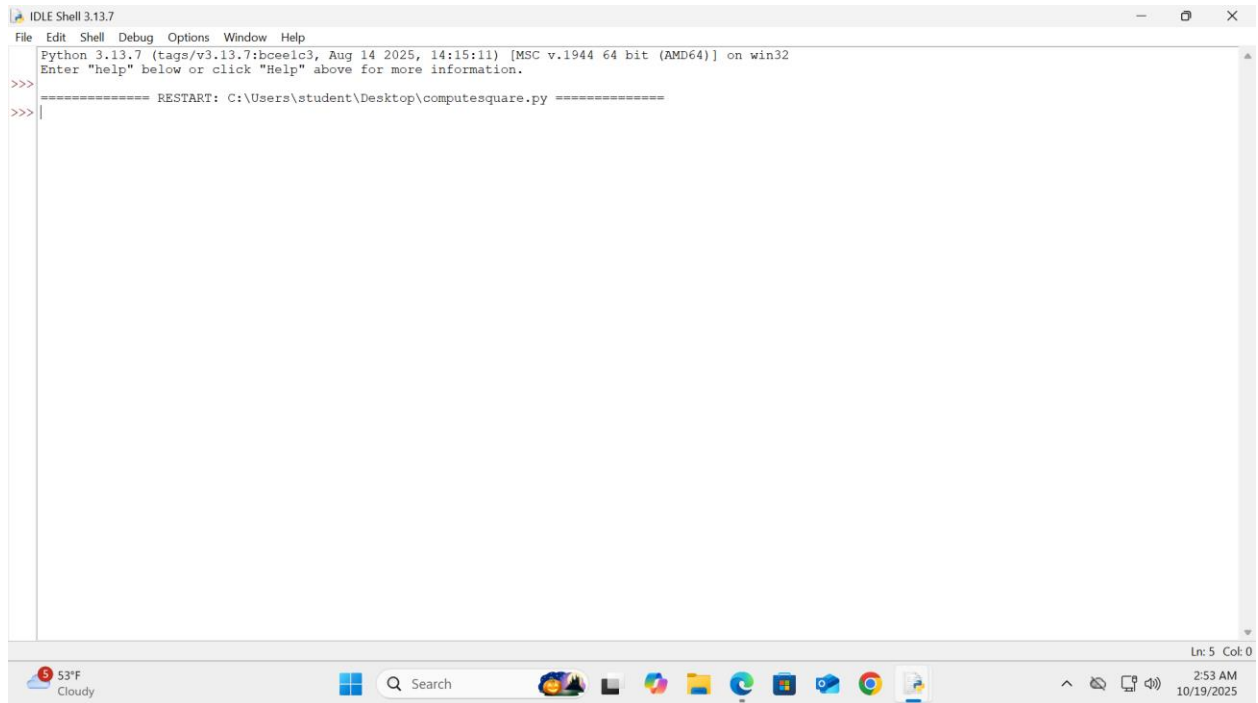


```
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>> def odd(x):
...     """Returns True if x is odd or False otherwise."""
...     if x % 2 == 1:
...         return True
...     else:
...         return False
...
>>> odd(5)
True
>>> odd(6)
False
>>> |
```

Defining the Main function

Page 128

1. You may find that the code example block is easier to implement with a script than in the interactive shell. So put this code example block into a script and then execute it, catching the screen shot after implementation.



```
IDLE Shell 3.13.7
File Edit Shell Debug Options Window Help
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>>
>>> ===== RESTART: C:\Users\student\Desktop\computesquare.py =====
>>>
```

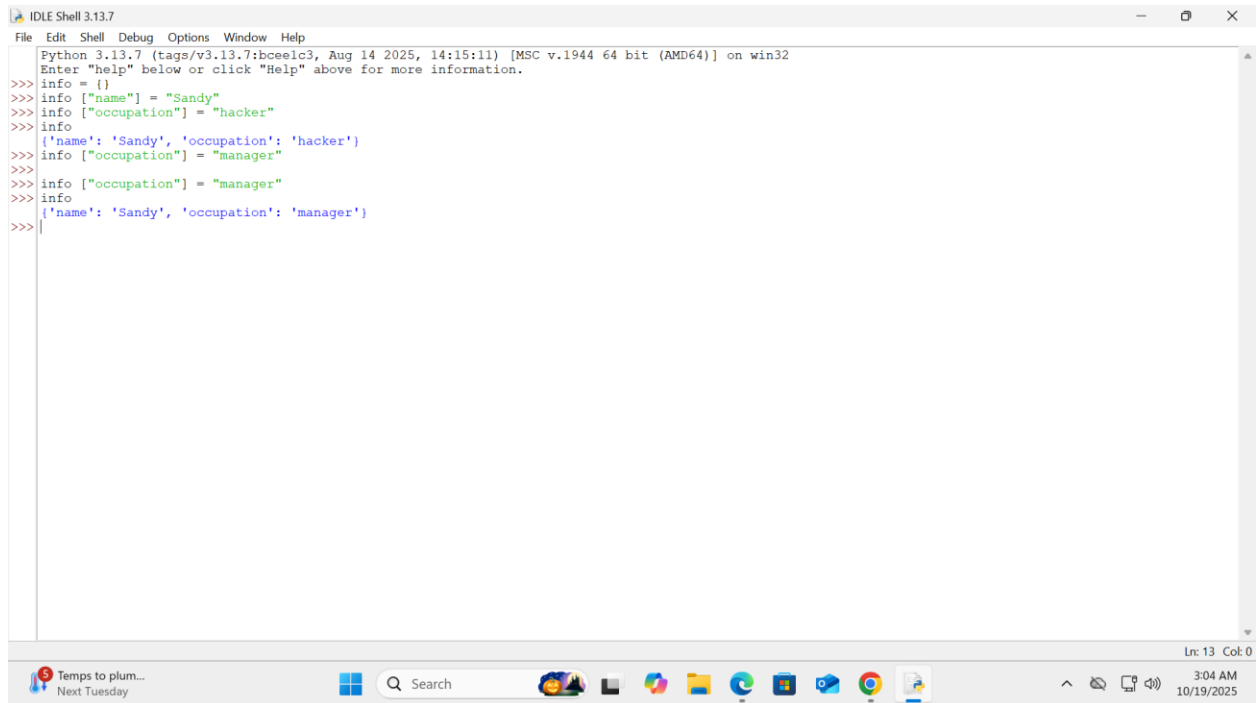
The Case study Generating Sentences is not one you need to catch screen shots of but you should go through it and use it as a learning tool.

Dictionaries

Adding Keys and Replacing Values

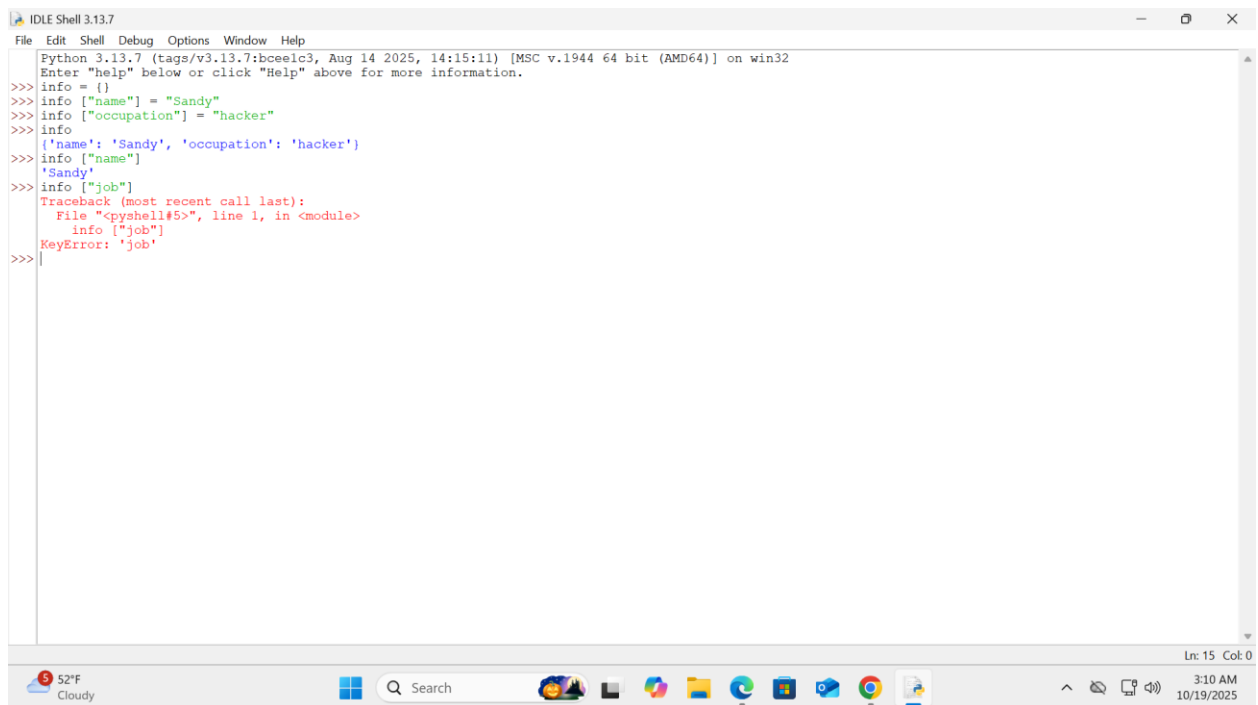
Page 132

1. At the bottom of the page, there is code example block of a few lines and then another line after some text. They are all together 1 block and you can include them all into a single screen shot.



```
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>> info = {}
>>> info["name"] = "Sandy"
>>> info["occupation"] = "hacker"
>>> info
{'name': 'Sandy', 'occupation': 'hacker'}
>>> info["occupation"] = "manager"
>>> info
{'name': 'Sandy', 'occupation': 'manager'}
>>> info["occupation"] = "manager"
>>> info
{'name': 'Sandy', 'occupation': 'manager'}
>>>
```

2. There is a small code example block at the end of the page. It shows an error condition. Grab a screenshot of the error you get. The code ends on the page also.

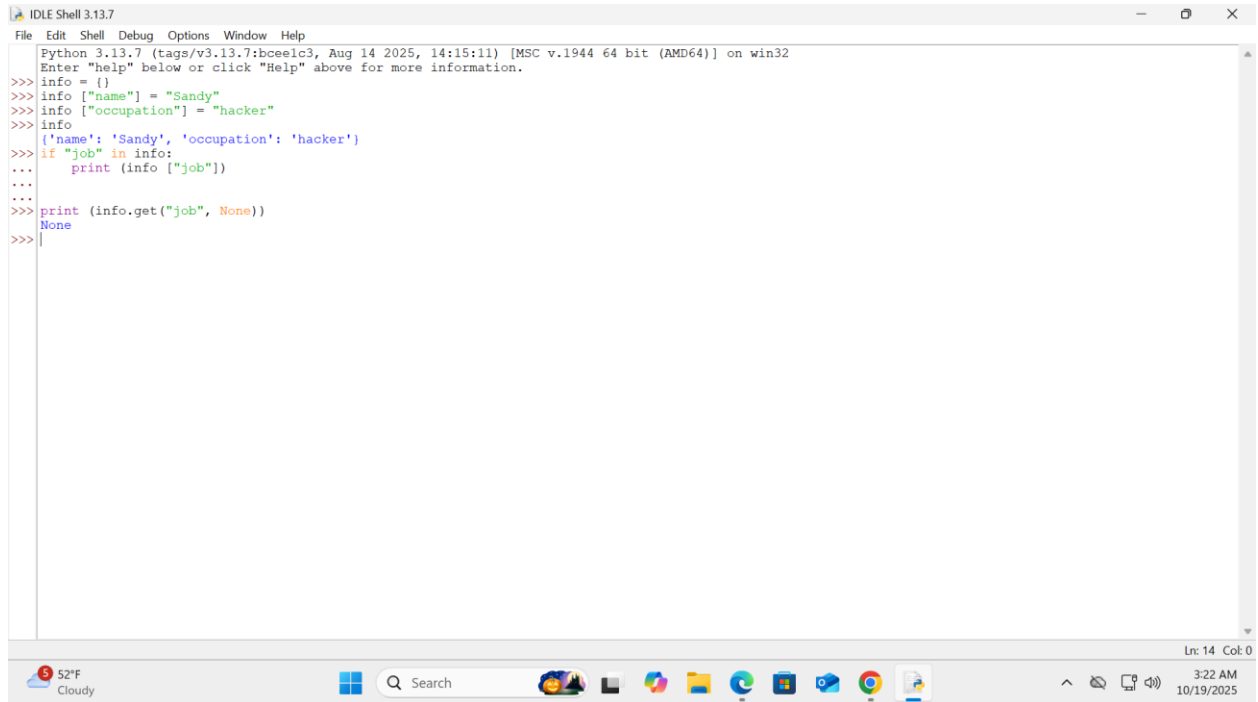


```
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>> info = {}
>>> info["name"] = "Sandy"
>>> info["occupation"] = "hacker"
>>> info
{'name': 'Sandy', 'occupation': 'hacker'}
>>> info["name"]
'Sandy'
>>> info["job"]
Traceback (most recent call last):
  File "<pyshell#5>", line 1, in <module>
    info["job"]
KeyError: 'job'
>>>
```

Accessing Values

Page 133

1. There are two small code example blocks at the top of the page.
Both can be placed into a single screen shot

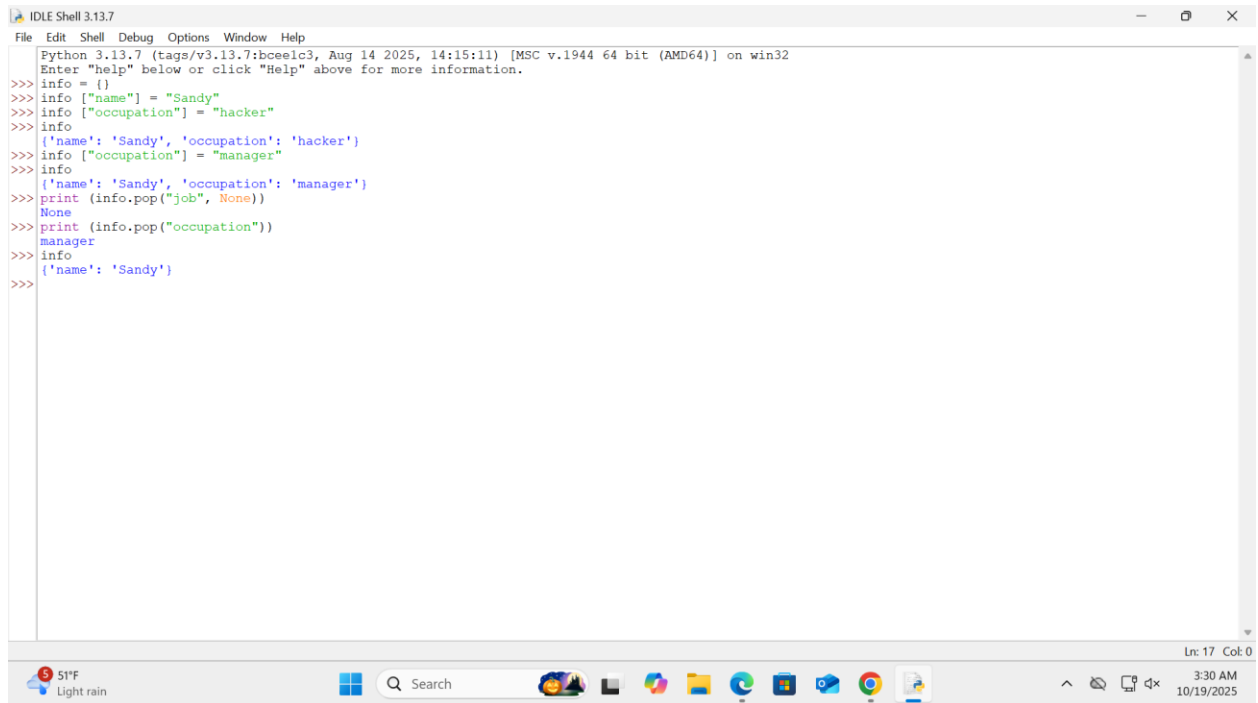


The screenshot shows a Python IDLE Shell window titled 'IDLE Shell 3.13.7'. The window contains the following Python code:

```
Python 3.13.7 (tags/v3.13.7:bce61c3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>> info = {}
>>> info["name"] = "Sandy"
>>> info["occupation"] = "hacker"
>>> info
{'name': 'Sandy', 'occupation': 'hacker'}
>>> if "job" in info:
...     print(info["job"])
...
>>> print(info.get("job", None))
None
>>>
```

The window's taskbar at the bottom shows the Windows Start button, a search bar, and several application icons. The system tray on the right indicates the temperature is 52°F, the weather is Cloudy, and the time is 3:22 AM on 10/19/2025.

2. In the middle of the page, there is a code example block about removing keys.



The screenshot shows an IDLE Shell window with the following Python code and output:

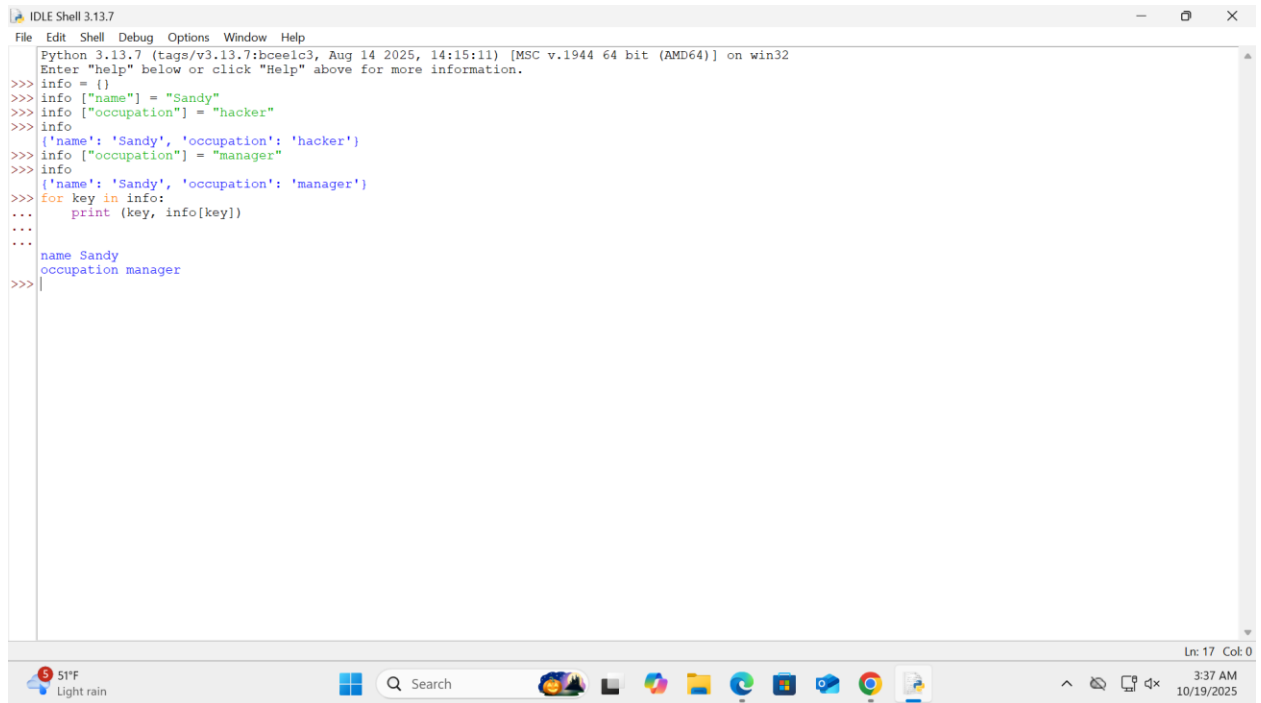
```
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>> info = {}
>>> info["name"] = "Sandy"
>>> info["occupation"] = "hacker"
>>> info
{'name': 'Sandy', 'occupation': 'hacker'}
>>> info["occupation"] = "manager"
>>> info
{'name': 'Sandy', 'occupation': 'manager'}
>>> print (info.pop("job", None))
None
>>> print (info.pop("occupation"))
manager
>>> info
{'name': 'Sandy'}
```

The window's status bar at the bottom shows the weather as 51°F with light rain, a search bar, and system icons for time (3:30 AM) and date (10/19/2025).

Traversing a Dictionary

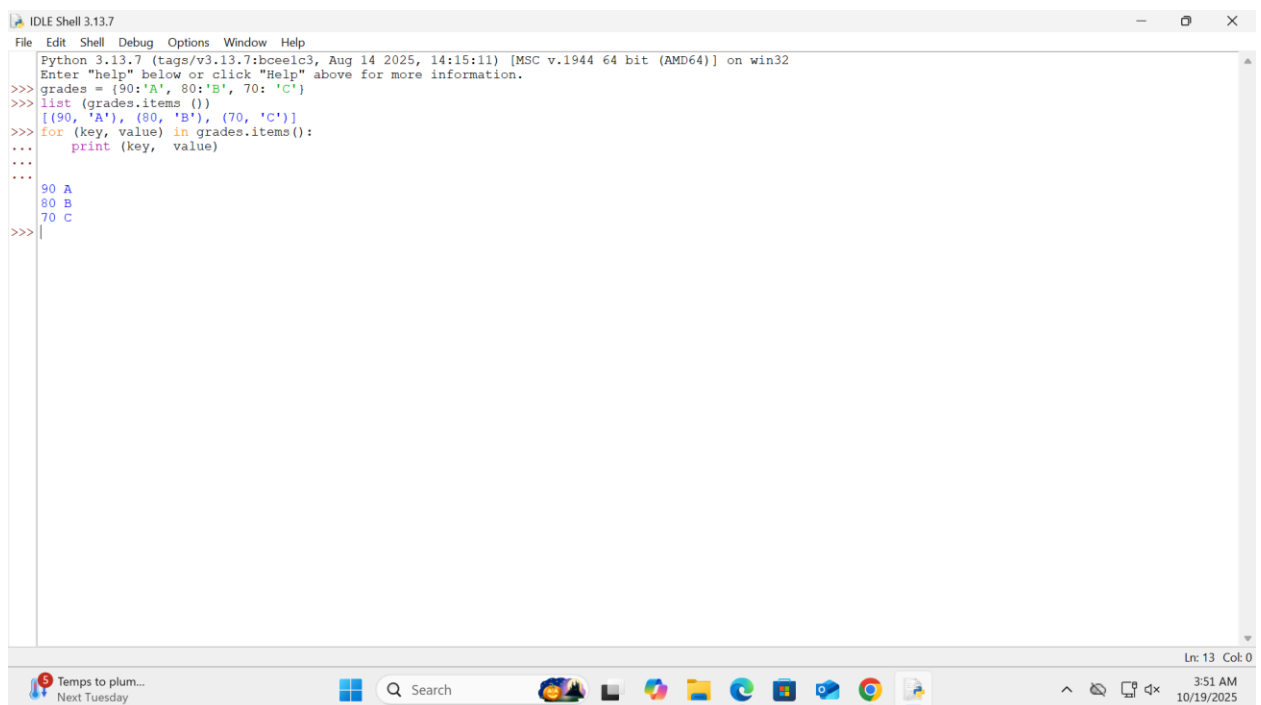
Page 133

1. The last code example blocks starts at the bottom of the page and spills over to the top of the next page. This can be grouped into a single screen shot if you would like.



```
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>> info = {}
>>> info["name"] = "Sandy"
>>> info["occupation"] = "hacker"
>>> info
{'name': 'Sandy', 'occupation': 'hacker'}
>>> info["occupation"] = "manager"
>>> info
{'name': 'Sandy', 'occupation': 'manager'}
>>> for key in info:
...     print (key, info[key])
...
...
name Sandy
occupation manager
>>>
```

2. There is a small code example block about 1/2 the way down that page.



```
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>> grades = {90:'A', 80:'B', 70:'C'}
>>> list(grades.items())
[(90, 'A'), (80, 'B'), (70, 'C')]
>>> for (key, value) in grades.items():
...     print (key, value)
...
...
90 A
80 B
70 C
>>>
```

Example: The Hexadecimal system revisited

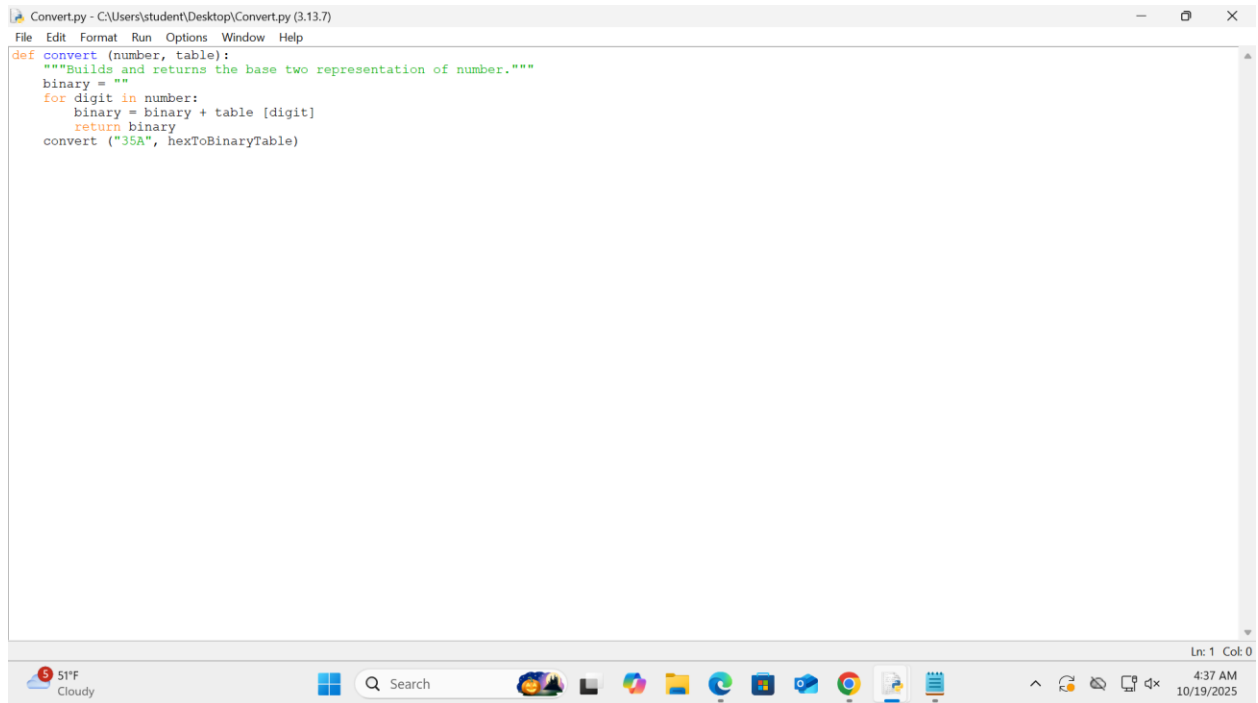
Page 134-135

1. The code example blocks here form a series of items that are interrelated. You may find this also easier to do in a script instead of interactive. If you do go the way of a script, be sure to run it and capture the output.

Example: Finding the Mode of a list of Values

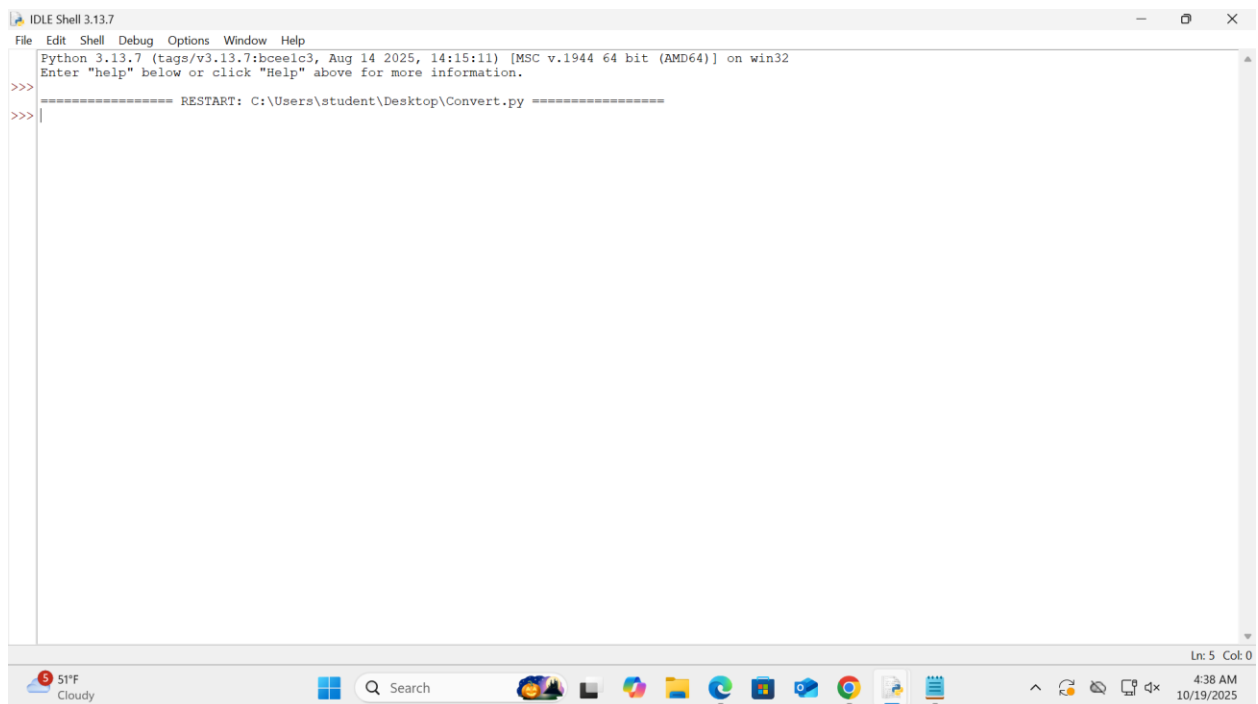
Page 135

1. The code example block at the bottom of the page and spills over to the top of the next page. Finish it up on this one and capture the screen shots as needed. This one too may be one you prefer to do in a script instead of the interactive mode.



```
Convert.py - C:\Users\student\Desktop\Convert.py (3.13.7)
File Edit Format Run Options Window Help
def convert(number, table):
    """Builds and returns the base two representation of number."""
    binary = ""
    for digit in number:
        binary = binary + table [digit]
    return binary
convert ("35A", hexToBinaryTable)
```

Ln: 1 Col: 0
4:37 AM
10/19/2025



```
IDLE Shell 3.13.7
File Edit Shell Debug Options Window Help
Python 3.13.7 (tags/v3.13.7:bce1c3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>>
===== RESTART: C:\Users\student\Desktop\Convert.py =====
>>>
|
```

Ln: 5 Col: 0
4:38 AM
10/19/2025

The Case Study on Psychotherapy is not included for grabbing screen shots but you should read through for learning opportunities.