Stellina Ao

CS 3035-05/06

Dr. Kaur

27 September 2022

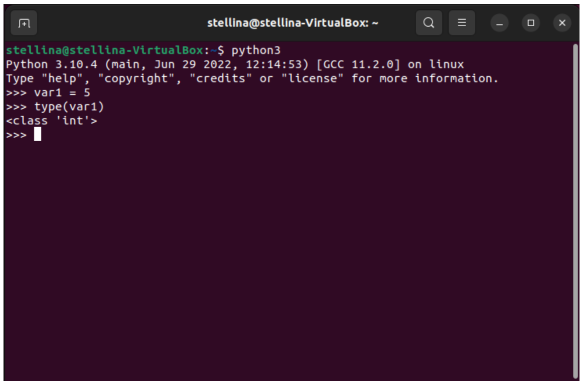
Lab 11

2.1) var1 = 5

2.2) type(var1)

2.2.1) What is the data type of variable ‘var1’?

The data type is an integer.



```
stellina@stellina-VirtualBox:~$ python3
Python 3.10.4 (main, Jun 29 2022, 12:14:53) [GCC 11.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> var1 = 5
>>> type(var1)
<class 'int'>
>>> 
```

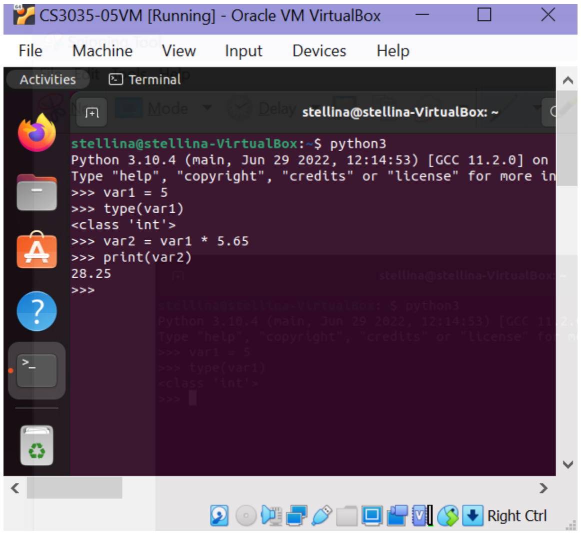
2.2.2) What type binding does Python use?

Python uses dynamic type binding.

2.3) var2 = var1*5.65 → print(var2)

2.3.1) What is the value of var2?

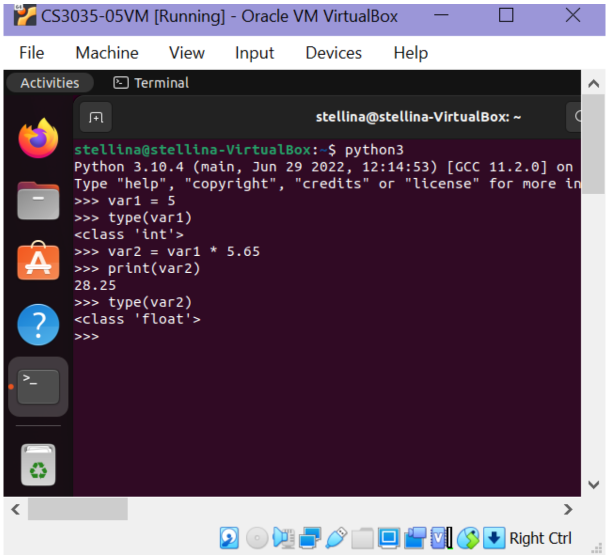
The value of var2 is 28.25.


CS3035-05VM [Running] - Oracle VM VirtualBox
File
Machine
View
Input
Devices
Help
Activities
Terminal
stellina@stellina-VirtualBox: ~
Right Ctrl

```
stellina@stellina-VirtualBox:~$ python3
Python 3.10.4 (main, Jun 29 2022, 12:14:53) [GCC 11.2.0] on
Type "help", "copyright", "credits" or "license" for more in
>>> var1 = 5
>>> type(var1)
<class 'int'>
>>> var2 = var1 * 5.65
>>> print(var2)
28.25
>>>
```

2.3.2) What is the data type of var2?

The data type of var2 is float.


CS3035-05VM [Running] - Oracle VM VirtualBox
File
Machine
View
Input
Devices
Help
Activities
Terminal
stellina@stellina-VirtualBox: ~
Right Ctrl

```
stellina@stellina-VirtualBox:~$ python3
Python 3.10.4 (main, Jun 29 2022, 12:14:53) [GCC 11.2.0] on
Type "help", "copyright", "credits" or "license" for more in
>>> var1 = 5
>>> type(var1)
<class 'int'>
>>> var2 = var1 * 5.65
>>> print(var2)
28.25
>>> type(var2)
<class 'float'>
>>>
```

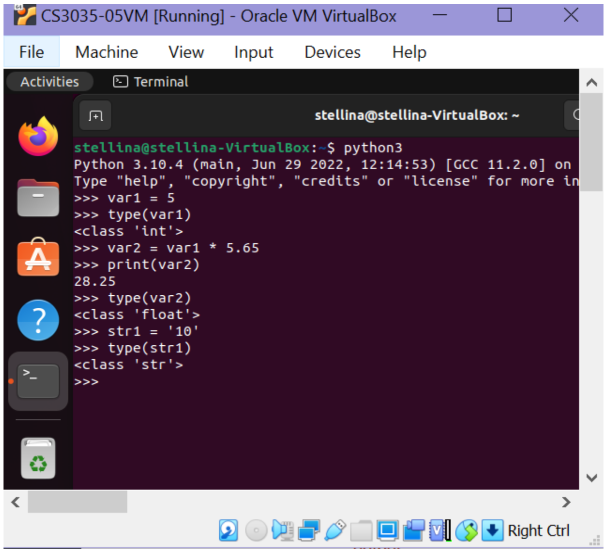
2.3.3) What data type conversion characteristics in Python do different types indicate?

This is characteristic of coercion, a feature wherein variables can be implicitly converted to legal types.

2.4) str1 = ‘10’

2.4.1) What is the data type of str1?

The data type of str1 is string.

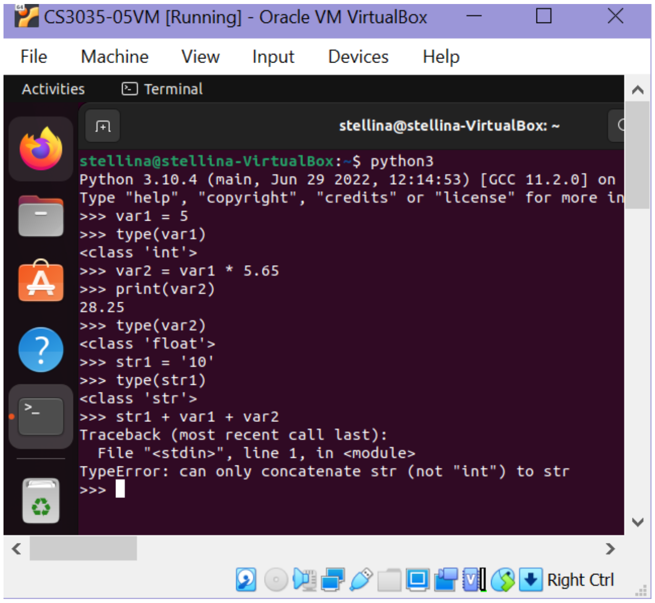

CS3035-05VM [Running] - Oracle VM VirtualBox
File
Machine
View
Input
Devices
Help
Activities
Terminal
stellina@stellina-VirtualBox: ~
Right Ctrl

```
stellina@stellina-VirtualBox:~$ python3
Python 3.10.4 (main, Jun 29 2022, 12:14:53) [GCC 11.2.0] on
Type "help", "copyright", "credits" or "license" for more in
>>> var1 = 5
>>> type(var1)
<class 'int'>
>>> var2 = var1 * 5.65
>>> print(var2)
28.25
>>> type(var2)
<class 'float'>
>>> str1 = '10'
>>> type(str1)
<class 'str'>
>>>
```

2.5) str1 + var1 + var2

2.5.1) What is the output?

There is a type error, as Python is unable to add the differing types.


CS3035-05VM [Running] - Oracle VM VirtualBox
File
Machine
View
Input
Devices
Help
Activities
Terminal
stellina@stellina-VirtualBox: ~
Right Ctrl

```
stellina@stellina-VirtualBox:~$ python3
Python 3.10.4 (main, Jun 29 2022, 12:14:53) [GCC 11.2.0] on
Type "help", "copyright", "credits" or "license" for more in
>>> var1 = 5
>>> type(var1)
<class 'int'>
>>> var2 = var1 * 5.65
>>> print(var2)
28.25
>>> type(var2)
<class 'float'>
>>> str1 = '10'
>>> type(str1)
<class 'str'>
>>> str1 + var1 + var2
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: can only concatenate str (not "int") to str
>>> 
```

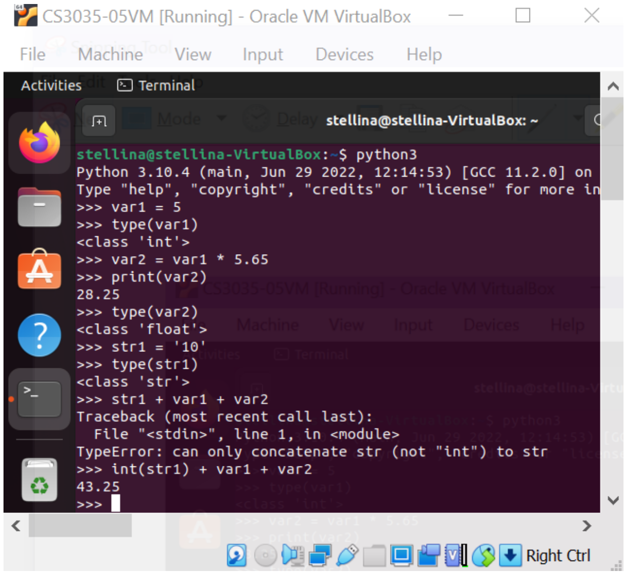
2.5.2) Which activity performed by the interpreter results in this output?

Python’s dynamic type checking detects this error and produces the output.

2.6) int(str1) + var1 + var2

2.6.1) What is the output?

The output was 43.25, the correct sum of 5, 28,25, and 10.


CS3035-05VM [Running] - Oracle VM VirtualBox
File
Machine
View
Input
Devices
Help
Activities
Terminal
stellina@stellina-VirtualBox: ~
Right Ctrl

```
stellina@stellina-VirtualBox:~$ python3
Python 3.10.4 (main, Jun 29 2022, 12:14:53) [GCC 11.2.0] on
Type "help", "copyright", "credits" or "license" for more in
>>> var1 = 5
>>> type(var1)
<class 'int'>
>>> var2 = var1 * 5.65
>>> print(var2)
28.25
>>> type(var2)
<class 'float'>
>>> str1 = '10'
>>> type(str1)
<class 'str'>
>>> str1 + var1 + var2
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: can only concatenate str (not "int") to str
>>> int(str1) + var1 + var2
43.25
>>> 
```

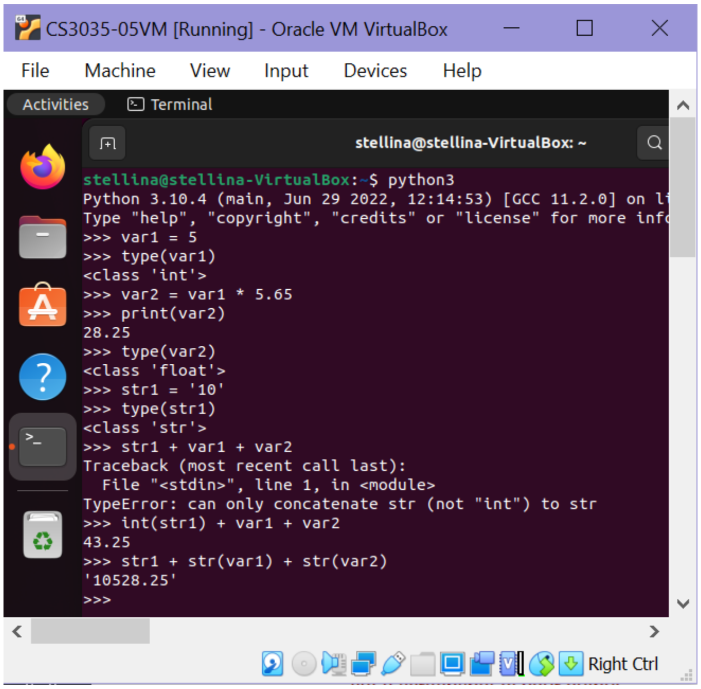
2.6.2) Why was there no error?

Python was able to recognize str1 as an int (due to our explicit type casting) and is also able to perform operations with integers and floats. So, this operation is valid.

2.7) str1 + str(var1) + str(var2)

2.7.1) What is the output?

The output is the string concatenation of 10, 5, and 28.25.


CS3035-05VM [Running] - Oracle VM VirtualBox
File
Machine
View
Input
Devices
Help
Activities
Terminal
stellina@stellina-VirtualBox: ~
Right Ctrl

```
stellina@stellina-VirtualBox:~$ python3
Python 3.10.4 (main, Jun 29 2022, 12:14:53) [GCC 11.2.0] on li
Type "help", "copyright", "credits" or "license" for more info
>>> var1 = 5
>>> type(var1)
<class 'int'>
>>> var2 = var1 * 5.65
>>> print(var2)
28.25
>>> type(var2)
<class 'float'>
>>> str1 = '10'
>>> type(str1)
<class 'str'>
>>> str1 + var1 + var2
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: can only concatenate str (not "int") to str
>>> int(str1) + var1 + var2
43.25
>>> str1 + str(var1) + str(var2)
'10528.25'
>>>
```

2.7.2) What operation does this expression seem to be doing?

The expression casts all non-string variables to strings, effectively rendering the '+' operator a string concatenation operator rather than an addition operator.