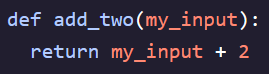
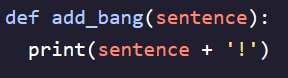
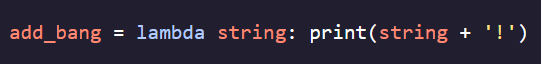
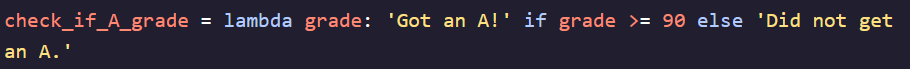
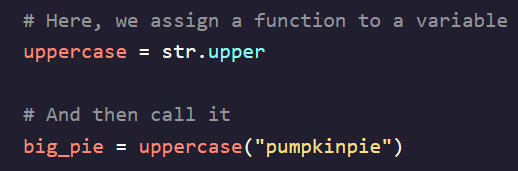
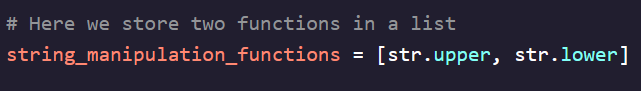
**Lambda Functions:**

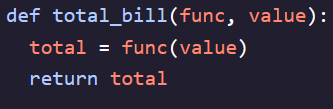
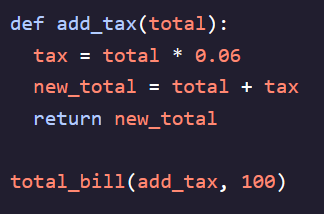
- Also known as an *anonymous function*, is a one-line shorthand for function  
- Use the *lambda* keyword to declare that this is a *lambda function*- The preferred way of creating one line functions as they improve readability – Not useful for functions that are more than one line  
- Primarily used where code reuse is not the primary objective  
   
 

- Can make them a little more complex  


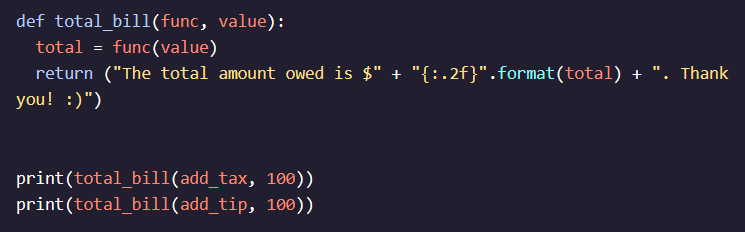
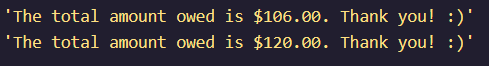
**Higher-Order Functions:**

- All functions are treated as *First-Class Objects* which have four important characteristics:  
1. Can be stored as variables  
2. Can be passed as arguments to a function  
3. Can be returned by a function  
4. Can be stored in a data structure (list, dictionary, tuple, etc)  
 

- *Higher-Order Functions* operate on other functions via arguments or via return values, they do one or both of the following:  
1. Accept a function as an argument  
2. Have a return value that is a function

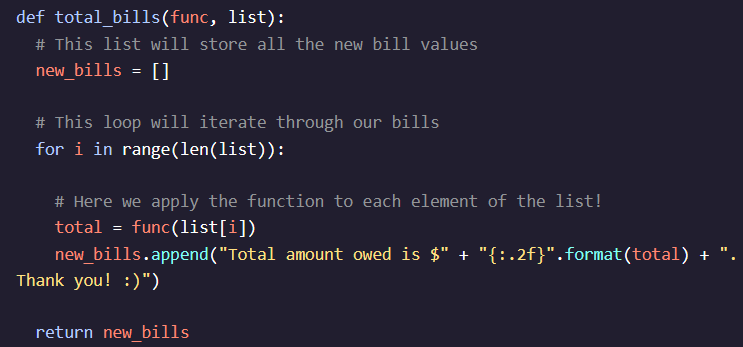
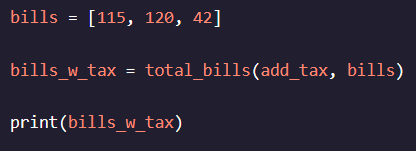
**Functions as Arguments:**  
 

- Takes two arguments: *func, value*   
- When called, *total\_bill* applies *function* to the *value* and returns result

"{:.2f}"

- is a method to format floating point numbers  
- {} denotes a placeholder for the value passed in later  
- : indicates start of format specifier  
- .2f specifies the floating point number should be formatted with 2 decimal places

**Functions as Return Values:**

- *make\_box\_volume\_function()* is the higher-order function that takes a height as an argument and returns a new function that calculates the volume of any box with that height when it is passed the length and the width of the box as well

