1. Define a function called greet that takes a single argument name and prints a greeting. Test the function by calling it with a few different values of name.
2. Define a function called add that takes two arguments x and y and returns their sum. Test the function by calling it with a few different values of x and y.
3. Define a function called is_even that takes a single argument x and returns True if x is even and False if x is odd. Test the function by calling it with a few different values of x.
4. Define a function called sum_all that takes a variable number of arguments and returns the sum of all of them. Test the function by calling it with a few different numbers of arguments.
5. Define a function called greet_all that takes a single argument names that is a list of names, and prints a greeting to each name. Test the function by calling it with a few different lists of names.
6. Define a function called average that takes a single argument values that is a list of numbers, and returns the average of all of them. Test the function by calling it with a few different lists of numbers.
7. Define a function called count_vowels that takes a single argument string and returns the number of vowels in the string. Test the function by calling it with a few different strings.
8. Define a function called reverse that takes a single argument string and returns the string with its characters reversed. Test the function by calling it with a few different strings.
9. Define a function called sort that takes a single argument values that is a list of numbers, and returns the list sorted in ascending order. Test the function by calling it with a few different lists of numbers.
10. Define a function called is_palindrome that takes a single argument string and returns True if the string is a palindrome (meaning it is the same forwards and backwards) and False otherwise. Test the function by calling it with a few different strings.
11. Define a function called max that takes a single argument values that is a list of numbers, and returns the maximum value in the list. Test the function by calling it with a few different lists of numbers.
12. Define a function called min that takes a single argument values that is a list of numbers, and returns the minimum value in the list. Test the function by calling it with a few different lists of numbers.

13. Define a function called count that takes a single argument values that is a list of numbers, and a second argument target that is a number, and returns the number of times that target appears in the list. Test the function by calling it with a few different lists of numbers and values of target.
14. Define a function called find that takes a single argument values that is a list of numbers, and a second argument target that is a number, and returns the index at which target appears in the list, or None if it does not appear. Test the function by calling it with a few different lists of numbers and values of target.
15. Define a function called remove that takes a single argument values that is a list of numbers, and a second argument target that is a number, and removes all occurrences of target from the list. Test the function by calling it with a few different lists of numbers and values of target.
16. Define a function called is_sorted that takes a single argument values that is a list of numbers, and returns True if the list is sorted in ascending order and False otherwise. Test the function by calling it with a few different lists of numbers.
17. Define a function called is_anagram that takes two arguments string1 and string2, and returns True if the two strings are anagrams (meaning they contain the same letters) and False otherwise. Test the function by calling it with a few different pairs of strings.
18. Define a function called fizzbuzz that takes a single argument n and prints the numbers from 1 to n, replacing multiples of 3 with "Fizz", multiples of 5 with "Buzz", and multiples of both 3 and 5 with "FizzBuzz". Test the function by calling it with a few different values of n.
19. Define a function called factorial that takes a single argument n and returns the factorial of n (the product of all the numbers from 1 to n). Test the function by calling it with a few different values of n.
20. Define a function called fibonacci that takes a single argument n and returns the first n numbers in the Fibonacci sequence (where each number is the sum of the previous two). Test the function by calling it with a few different values of n.