University of Strathclyde Glasgow

Python: Conditions and loops

1. Write a program that uses a `for` loop to print all of the keys that are stored in a dictionary.
2. Write a program that sums all of the integers between 5 and 15. The program should print the result of the addition.
3. Write a program that contains an integer variable. Write a `for` loop that starts from 1 and continues to the value of the integer variable + 1. Multiply each of the numbers together. Print the result of the multiplication.
4. Write a program that converts the string values `"one"`, `"two"`, `"three"`, `"four"`, `"five"`, `"six"`, `"seven"`, `"eight"` and `"nine"` into the corresponding integer value. For example, the program should print `1`, when a string variable contains `"one"`. The program should use a dictionary to solve this problem and check if the string is a dictionary key before using it to lookup the value.
5. Write a program that converts a string into an integer, by casting the string to an integer. If the string is not an integer, then the program should print an error message. If the string is an integer, then the program should add 10 to the integer value and print it.

   The function should use `try` and `except` to catch the `ValueError` exception that is thrown if the string is not an integer.
6. Write a program that prints all numbers between 10 and 26 that are not multiples of 2 and 3. The operator `%` can be used to test if a number is a multiple of another number. For example, `i % 2` returns 0 if `i` is a multiple of 2.