# **Name:** Karin Sinkevitch

**Date:** August 10, 2020

**Course:** IT FDN 110A

**Assignment:** 05

**GitHub URL:** <https://github.com/KARSINK22/IntroToProg-Python>

# **Lists and Dictionaries, Error Handling, Organizing Code, and Script Templates**

# Introduction

This document provides a brief introduction to using Python lists and dictionaries as well as information on basic error handling, organizing code, and creating script templates. Reference information can be used to learn more details and see examples of the topics covered. An example script is described to show how these methods can be applied.

# Lists and Dictionaries

# Lists

Lists are like arrays and may contain data types like Integers, Strings, as well as Objects. Lists are mutable, and can be altered even after their creation. When creating a list a bracket is used. Common list methods are listed in Figure 1. (Reference: <https://www.geeksforgeeks.org/difference-between-list-and-dictionary-in-python/> and “Python Programming for the Absolute Beginner, 3rd Edition” by Michael Dawson).

Table 5.1: Selected List Methods 
append(va/ue) 
sort () 
reverse () 
count(value) 
index(value) 
insert(i, 
value) 
POP ( [i] ) 
remove ( value) 
Adds value to end of a list. 
Sorts the elements, smallest value first. Optionally, you can pass a Boolean value to the 
parameter reverse. If you pass True, the list will be sorted with the largest value first. 
Reverses the order of a list. 
Returns the number of occurrences of value. 
Returns the first position number of where value occurs. 
Inserts value at position i. 
Returns value at position i and removes value from the list. Providing the position number i is 
optional. Without it, the last element in the list is removed and returned. 
Removes the first occurrence Of value from the list. 

Figure 1 Common List Methods

# Dictionaries

A dictionary is an unordered collection of data values which unlike other Data Types that hold only single value as an element, holds a key:value pair. When creating a dictionary a curly brace is used. Common list methods are listed in Figure 2. (Reference: <https://www.geeksforgeeks.org/difference-between-list-and-dictionary-in-python/> and “Python Programming for the Absolute Beginner, 3rd Edition” by Michael Dawson).

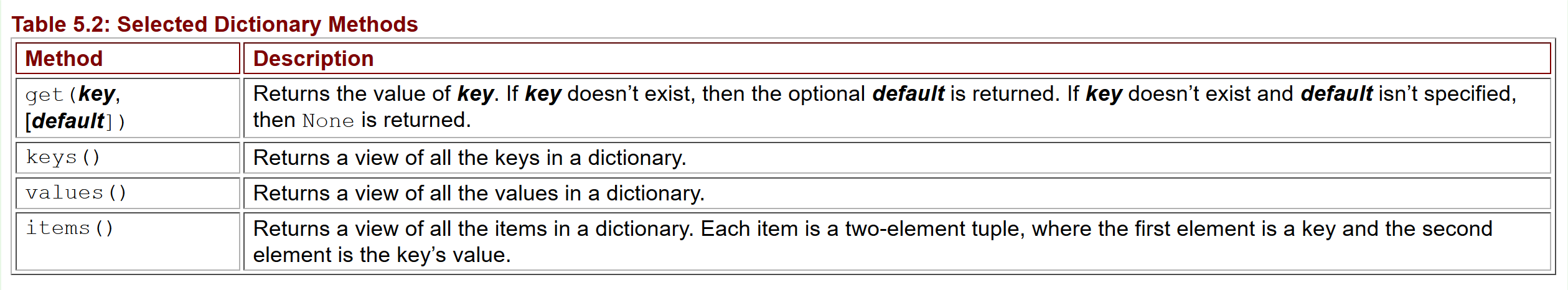


Figure 2 Common Dictionary Methods

# List of Dictionaries

A common way of organizing data in Python is by using a list of dictionaries. A great discussion on this with examples can be found in the references. (References: <https://www.w3resource.com/python-exercises/dictionary/python-data-type-dictionary-exercise-23.php> and https://www.youtube.com/watch?v=P5wOsnPjn6Y&feature=youtu.be)

# Error Handling

Simple error handling can be performed using Try – Catch. The code you want to “try” is indented under the try block and if there is an error, the code under the catch block is executed. This allows for more information to be provided to the user and can also be used to make a decision without the code stopping with an error. (Reference: https://wiki.python.org/moin/HandlingExceptions).

# Organizing Code

A basis of clear code is the division of its various behaviors into small, manageable pieces. Short segments of code with clear intent as well as separating them by concern is an effective approach. A concern is a distinct behavior or piece of knowledge your software deals with. It is recommended to section code into data, processing, and presentation or input/output. (References: <https://livebook.manning.com/book/code-like-a-pro/chapter-2/v-5/4> and [Randal](https://www.geeksforgeeks.org/loops-in-python/) Root “Mod5PythonProgrammingNotes”).

# Script Templates

PyCharm contains functionality to allow users to create script templates.The steps for creating templates is straightforward and can be found in the reference provided. (Reference: https://www.jetbrains.com/help/pycharm/using-file-and-code-templates.html).

# Create Python Script “Assigment05\_Starter.py”

This section describes how to create a script called “Assigment05\_Starter.py”. The script reads in an existing to do list called “ToDoList.txt” which is a comma separated file with a task and a priority. The list is read into a list of dictionaries. Each dictionary contains two values: task and priority. Try-except error handling is used when reading the file in case the file does not exist or there is an error when reading it. The script then provides a menu to the user to view or manipulate the list (Figure 3).

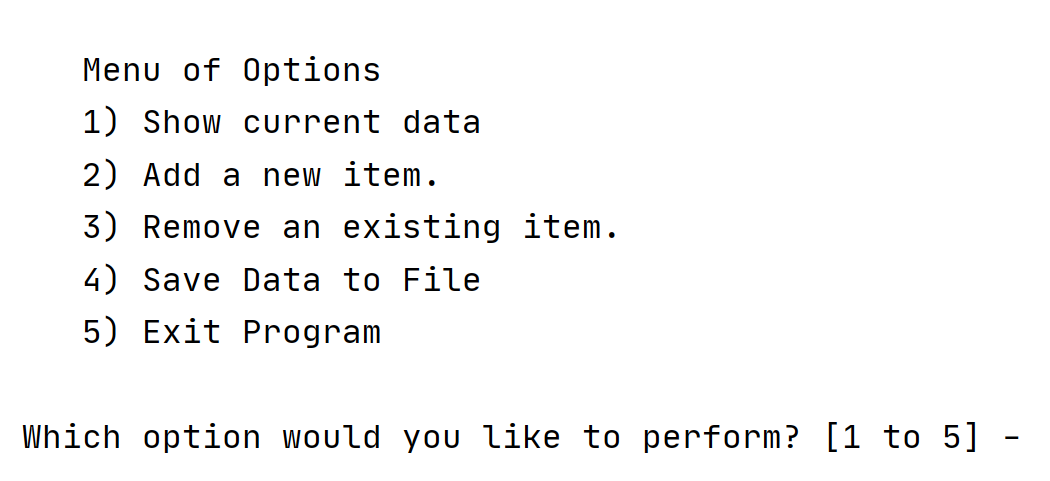


Figure 3 Assignment05\_Starter.py Menu

If the user selects “1”, the existing data is printed to the screen (Figure 4). This is accomplished by using a for loop to print each dictionary element or row in the list.

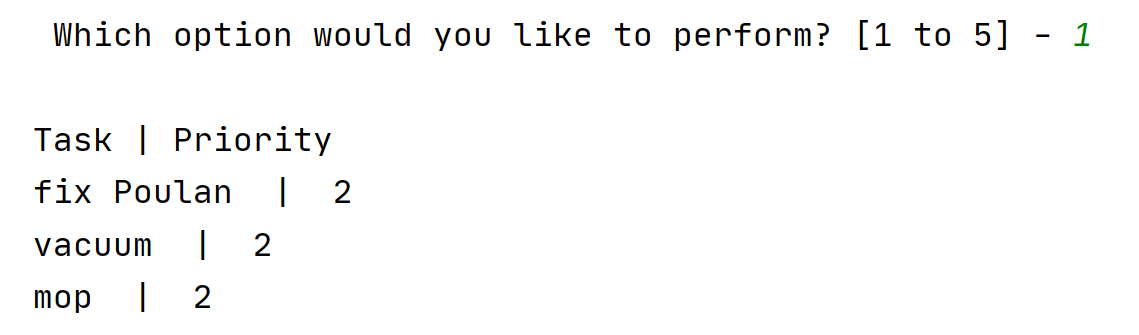


Figure 4 Assignment05\_Starter.py Show Current Data

If the user selects “2”, the user is asked to enter a new task and priority (Figure 5). This input is added to the list as a new dictionary element using the list method “append”.

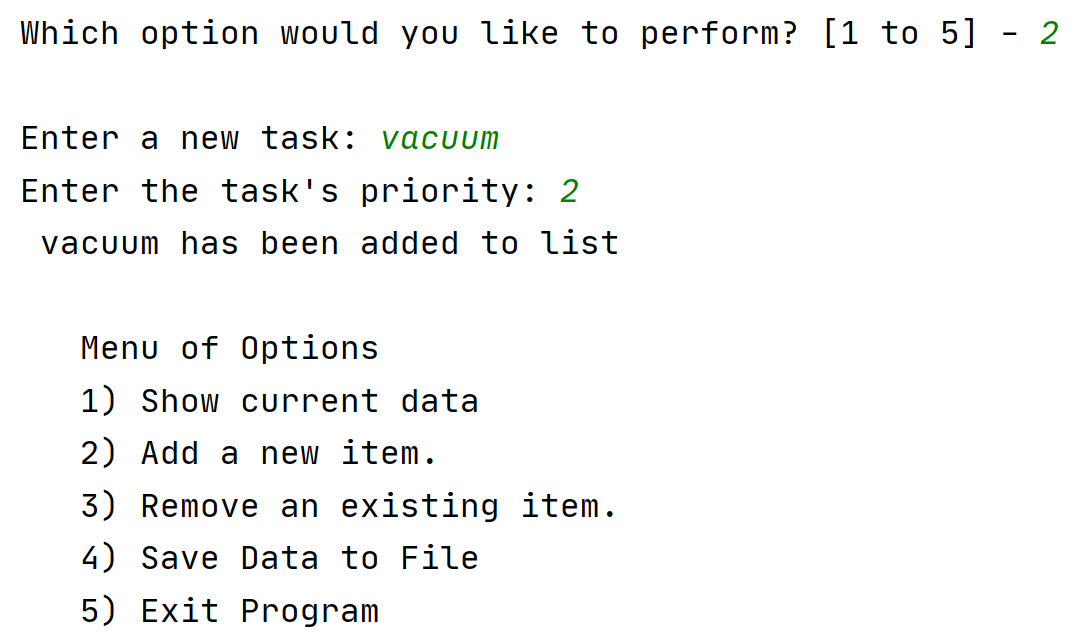


Figure 5 Assignment05\_Starter.py Add New Item

If the user selects “3”, the user is asked to input the item in the list to be remove. The script uses a for loop to step through the list to find the dictionary task value that matches the input (Figure 6). If the item is found, it is removed using the list method “remove”. If the item is not found, the user is notified that the item was not found in the list.

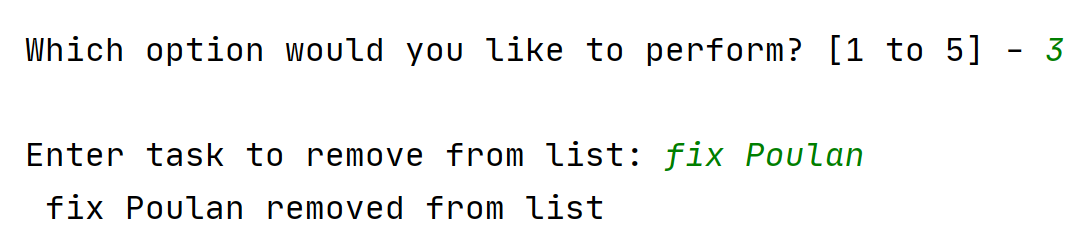


Figure 6 Assignment05\_Starter.py Remove an Item

If the user selects “4”, the list is saved to the file. The existing data is overwritten by using the “w” option of the file IO method. Try-except error handling is used in case there is an error saving the file.

If the user selects “5”, the program exits.

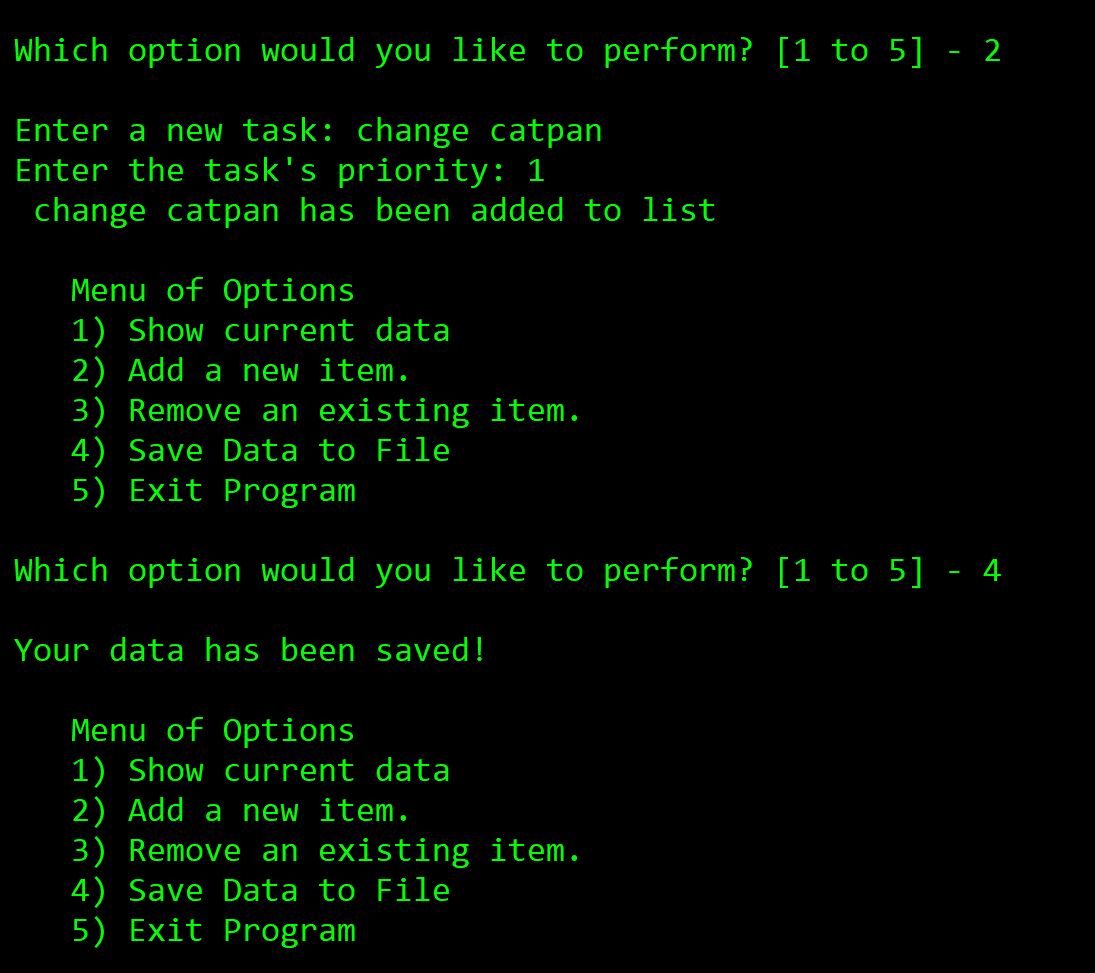


Figure 7 Assignment05\_Starter.py Execution in Console

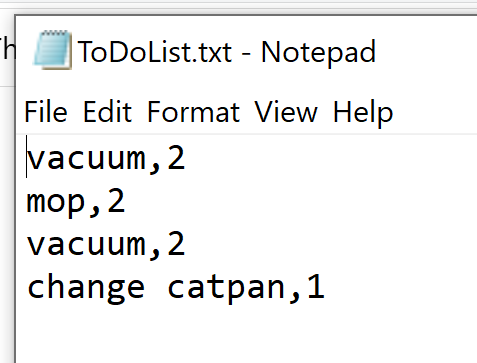


Figure 8 ToDoList.txt

# Summary

Using a list of dictionaries to capture lists of data is an immensely powerful capability in Python. Basic error handling with try-except can provide the user with more information on the problem. Organizing code will allow others to read the code easier and creating script templates can make starting a new script easier. More information on the subject covered can be found in the references.