

Lab 6.03 - Dictionaries Storing Lists

In this lab your job is to create a week-long to-do list using a Python dictionary. Each key in the dictionary is a day of the week. Each associated value is a list of items to do that day.

The program repeatedly asks the user what action they wish to take (**add** or **get**).

- If the user enters **get**, the program asks for a day of the week, and then returns the to-do list for that day.
- If the user enters **add**, the program asks for a day of the week, then asks for a new item, then adds it to the specified list.
- If a user tries to add an item that already exists on the list for that day, the program rejects the request.
- At the start of the program the dictionary should be totally empty (containing no keys and no values).

Example

Here's an example. The program output is shown in bold text, the user input in regular text.

```
>>>python3 daily_to_do_list.py
What would you like to do ('add' or 'get')?
add
What day?
Friday
What would you like to add to Fridays to-do list?
practice clarinet
What would you like to do ('add' or 'get')?
get
What day?
Friday
You have to practice clarinet.
What would you like to do('add' or 'get')?
```

Bonus

It's a bit tedious for the user to have to type in three different lines to add an item to a to-do list. Use `split()` to allow the user to input **add Friday watch tv and relax**.

Create a variation of the program that doesn't allow any duplicates across any of the days. Make sure when you add a to-do item that it doesn't exist in the to-do lists of any of the days before adding.