

```

import os, pickle

# user input to build a dictionary that includes a list
# also accessing existing element, modifying, deleting

def open_file():
    try:
        file_object = open('animalsx.pydata', 'rb')
        animalsx = pickle.load(file_object)
        file_object.close()
    except:
        # return an empty dictionary not an array!
        animalsx = {}
    return animalsx

def close_file():
    try:
        file_object = open('animalsx.pydata', 'wb')
        pickle.dump(animalsx, file_object)
        file_object.close()
        print("\nSuccess! Your rando animal list has been saved\n")
    except Exception as e:
        print(e)
        print("\n\tSorry, something went wrong and the list is not saved")

def print_list():
    if len(animalsx) > 0:
        for animal in animalsx:
            print("\nAnimal Details: %s" % animal)
            # index & enumerate lets you keep track of which item in the you have
            for index,detail in enumerate(animalsx[animal]):
                if (index == 0):
                    print("\tname: %s" % detail)
                elif (index == 1):
                    print("\tTcolor: %s" % detail)
                    # no need to check index if this is the last case
                else:
                    print("\tTemperament: %s" % detail)
    else:
        print("There is nothing here yet")

def add_animal():
    a_type = input("\nType of animal to add? ")

```

```
a_name = input("What is the name of this animal? ")
a_name = a_name.lower()
a_color = input("What color is the animal? ")
a_temp = input("What is the animal's temperament? ")

# this updates the dictionary with a new entry: key + list of values
animalsx.update({a_type : [a_name, a_color, a_temp]})
print("%s has been added to the list" % a_type)
```

```
# if key found then make changes
```

```
def change_animal():
```

```
    a_type = input("What type of animal are you changing? ")
```

```
    if a_type in animalsx:
```

```
        a_name = input("What is the name of this animal? ")
```

```
        a_name = a_name.lower()
```

```
        a_color = input("What color is the animal? ")
```

```
        a_temp = input("What is the animal's temperament? ")
```

```
        animalsx.update({a_type : [a_name, a_color, a_temp]})
```

```
        print("%s was changed" % a_type)
```

```
        print("here")
```

```
    else:
```

```
        print("Sorry, that animal does not seem to be here. Check the list again.")
```

```
def remove_animal():
```

```
    a_type = input("What type of animal do you want to remove? ")
```

```
    if a_type in animalsx:
```

```
        del animalsx[a_type]
```

```
        print("%s was removed " % a_type)
```

```
    else:
```

```
        print("Sorry, that animal is not here. Check the list again.")
```

```
def print_menu():
```

```
    os.system('clear')
```

```
    print("1 to print the list")
```

```
    print("2 to add a new animal")
```

```
    print("3 to change an animal")
```

```
    print("4 to remove an animal")
```

```
    print("5 to quit")
```

```
# this would be a parent algorithm
```

```
# any of the functions called by this could be the children
```

```
def process_menu():
```

```
choice = input("What would you like to do? ")
print_menu()
if choice == '1':
    print_list()
elif choice == "2":
    add_animal()
elif choice == '3':
    change_animal()
elif choice == '4':
    remove_animal()
else:
    if choice != '5':
        print("What are you doing. That is not an option.")
    return choice
```

```
# main
# if file is there open and load else create empty dictionary
animalsx = open_file()
```

```
#
choice = ""
print_menu()
```

```
# this will keep running until user enters a '5'
while choice != '5':
    choice = process_menu()
```

```
# finish up
close_file()
```