Using zip - Python Toolbox Exercise

Exercise: Using zip

Another interesting function that you've learned is zip(), which takes any number of iterables and returns a zip object that is an iterator of tuples. If you wanted to print the values of a zip object, you can convert it into a list and then print it. Printing just a zip object will not return the values unless you unpack it first. In this exercise, you will explore this for yourself.

Three lists of strings are pre-loaded: mutants, aliases, and powers. First, you will use list() and the zip() function on these lists to generate a list of tuples. Then, you will create a zip object using zip(). Finally, you will unpack this zip object in a for loop to print the values in each tuple. Observe the different output generated by printing the list of tuples, then the zip object, and finally, the tuple values in the for loop.

Instructions:

- Using zip() with list(), create a list of tuples from the three lists mutants, aliases, and powers (in that order) and assign the result to mutant_data.
- Using zip(), create a zip object called mutant_zip from the three lists mutants, aliases, and powers.
- Complete the for loop by unpacking the zip object you created and printing the tuple values. Use value1, value2, value3 for the values from each of mutants, aliases, and powers, in that order.

Python Code

```
# Preloaded lists
mutants = ['charles xavier', 'bobby drake', 'kurt wagner', 'max eisenhardt',
'kitty pryde']
aliases = ['prof x', 'iceman', 'nightcrawler', 'magneto', 'shadowcat']
powers = ['telepathy', 'thermokinesis', 'teleportation', 'magnetokinesis',
'intangibility']

# Using zip() with list() to create a list of tuples
mutant_data = list(zip(mutants, aliases, powers))
print(mutant_data)

# Using zip() to create a zip object
mutant_zip = zip(mutants, aliases, powers)
print(mutant_zip)
```

```
# Unpacking the zip object and printing tuple values
for value1, value2, value3 in mutant_zip:
    print(value1, value2, value3)
```

Explanation

1. The zip() function is used to combine elements from the lists mutants, aliases, and powers into tuples, where each tuple contains one element from each of the lists.
2. The list() function converts the zip object into a list of tuples, which is printed using print(mutant_data).
3. A zip object mutant_zip is created but is not directly printable in a human-readable form.
4. The for loop unpacks each tuple in mutant_zip into three variables: value1, value2, and value3. These variables are then printed.

This demonstrates the functionality of zip() both as an iterator and when used with list().