Reading a CSV File

There are two ways to read a CSV file. You can use the csv module's **reader** function or you can use the **DictReader** class. We will look at both methods. But first, we need to get a CSV file so we have something to parse. We've already created a `data.csv` file for you (and uploaded it to Educative'e execution directory) with the following contents

```
first_name,last_name,address,city,state,zip_code
Tyrese,Hirthe,1404 Turner Ville,Strackeport,NY,19106-8813
Jules,Dicki,2410 Estella Cape Suite 061,Lake Nickolasville,ME,00621-7435
Dedric,Medhurst,6912 Dayna Shoal,Stiedemannberg,SC,43259-2273
```

Now let's look at some code

```python
import csv

def csv_reader(file_obj):
    """
    Read a csv file
    """
    reader = csv.reader(file_obj)
    for row in reader:
        print(" ".join(row))

if __name__ == "__main__":
    csv_path = "data.csv"
    with open(csv_path, "r") as f_obj:
        csv_reader(f_obj)
```

Let's take a moment to break this down a bit. First off, we have to actually import the **csv** module. Then we create a very simple function called **csv_reader** that accepts a file object. Inside the function, we pass the file object into the **csv.reader** function, which returns a reader object. The reader object allows iteration, much like a regular file object does. This lets us iterate over each row in the reader object and print out the line of data, minus the commas. This works because each row is a list and we can join each element in the list together, forming one long string.

Now that we understand the format of our csv file, let’s feed it into the **DictReader** class.

```
import csv

def csv_dict_reader(file_obj):
    """
    Read a CSV file using csv.DictReader
    """
    reader = csv.DictReader(file_obj, delimiter=',')
    for line in reader:
        print(line["first_name"]),
        print(line["last_name"])

if __name__ == "__main__":
    with open("data.csv") as f_obj:
        csv_dict_reader(f_obj)
```

In the example above, we open a file and pass the file object to our function as we did before. The function passes the file object to our DictReader class. We tell the DictReader that the delimiter is a comma. This isn’t actually required as the code will still work without that keyword argument. However, it’s a good idea to be explicit so you know what’s going on here. Next we loop over the reader object and discover that each line in the reader object is a dictionary. This makes printing out specific pieces of the line very easy.

Now we’re ready to learn how to write a csv file to disk.