Practice Quiz: Reading & Writing CSV Files

TOTAL POINTS 5

1. We're working with a list of flowers and some information about each one. The create_file function writes this information to a CSV file. The contents_of_file function reads this file into records and returns the information in a nicely formatted block. Fill in the gaps of the contents_of_file function to turn the data in the CSV file into a dictionary using DictReader. 1 point

```
import os
import csv

# Create a file with data in it
def create_file(filename):
  with open(filename, "w") as file:
    file.write("name,color,type\n")
    file.write("carnation,pink,annual\n")
    file.write("daffodil,yellow,perennial\n")
    file.write("iris,blue,perennial\n")
    file.write("poinsettia,red,perennial\n")
    file.write("sunflower,yellow,annual\n")

# Read the file contents and format the information about each row
def contents_of_file(filename):
  return_string = ""

  # Call the function to create the file
  create_file(filename)

  # Open the file
  with open(filename) as file:
    # Read the rows of the file into a dictionary
    file = csv.DictReader(file)
    # Process each item of the dictionary
    for row in file:
      return_string += "a {} {} is {}\n".format(row["color"], row["name"], r
  return return_string

#Call the function
print(contents_of_file("flowers.csv"))
```

Run Reset

```
a pink carnation is annual
a yellow daffodil is perennial
a blue iris is perennial
a red poinsettia is perennial
a yellow sunflower is annual
```

2. Using the CSV file of flowers again, fill in the gaps of the contents_of_file function to process the data without turning it into a dictionary. How do you skip over the header record with the field names? 1 point

```
import os
import csv

# Create a file with data in it
def create_file(filename):
  with open(filename, "w") as file:
    file.write("name,color,type\n")
    file.write("carnation,pink,annual\n")
    file.write("daffodil,yellow,perennial\n")
    file.write("iris,blue,perennial\n")
    file.write("poinsettia,red,perennial\n")
    file.write("sunflower,yellow,annual\n")

# Read the file contents and format the information about each row
def contents_of_file(filename):
  return_string = ""

  # Call the function to create the file
  create_file(filename)

  # Open the file
  with open(filename) as file:
    # Read the rows of the file
    rows = csv.reader(file)
    next(rows, None)
    # Process each row
    for row in rows:
      name, color, type = row
      # Format the return string for data rows only
      return_string += "a {} {} is {}\n".format(color, name, type)
  return return_string

#Call the function
print(contents_of_file("flowers.csv"))
```

Run Reset

```
a pink carnation is annual
a yellow daffodil is perennial
a blue iris is perennial
a red poinsettia is perennial
a yellow sunflower is annual
```

3. In order to use the writerows() function of DictWriter() to write a list of dictionaries to each line of a CSV file, what steps should we take? (Check all that apply) — 1 point

- [x] Create an instance of the DictWriter() class
- [x] Write the fieldnames parameter into the first row using writeheader()
- [x] Open the csv file using *with open*
- [ ] Import the OS module

4. Which of the following is true about unpacking values into variables when reading rows of a CSV file? (Check all that apply) — 1 point

- [x] We need the same amount of variables as there are columns of data in the CSV
- [x] Rows can be read using both csv.reader and csv.DictReader
- [x] An instance of the reader class must be created first
- [ ] The CSV file does not have to be explicitly opened

5. If we are analyzing a file's contents to correctly structure its data, what action are we performing on the file? — 1 point

- ( ) Writing
- ( ) Appending
- (•) Parsing
- ( ) Reading
