**Practice Quiz: Reading & Writing CSV Files**

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.

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

  \_\_\_

    # Read the rows of the file into a dictionary

    \_\_\_

    # Process each item of the dictionary

    for \_\_\_:

      return\_string += "a {} {} is {}\n".format(row["color"], row["name"], row["type"])

  return return\_string

#Call the function

print(contents\_of\_file("flowers.csv"))

solution:

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?  
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

  \_\_\_

    # Read the rows of the file

    rows = \_\_\_

    # Process each row

    for row in rows:

      \_\_\_ = row

      # Format the return string for data rows only

          return\_string += "a {} {} is {}\n".format(\_\_\_)

  return return\_string

#Call the function

print(contents\_of\_file("flowers.csv"))

solution:

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



Create an instance of the DictWriter() class



Write the fieldnames parameter into the first row using writeheader()



Open the csv file using *with open*



Import the OS module

4.Question 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



We need the same amount of variables as there are columns of data in the CSV



Rows can be read using both csv.reader and csv.DictReader



An instance of the reader class must be created first



The CSV file does not have to be explicitly opened

5.Question 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