TASK 6 **Implement various text file operations.**

Problem 1:
In a famous tech firm, an employee is asked **to copy the contents of N no.of files to another N no. of files**. Fortunately, the employee has good knowledge in python scripting. So he plans to automate the copy operation by creating a copy function using python file handling operations. Assume you are that employee and implement the copy function to achieve the task.
Example:

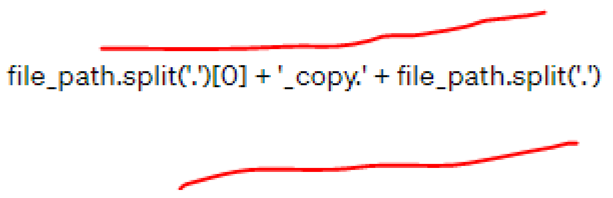
Input:
‘Asdfghjk’

File 1: (write contents from input to file 1)
Asdfghjk
Output: (copy the contents of file 1 to file 2)
File 2:
Asdfghjk

PROGRAM

```
def copy_files(file_paths):
    for file_path in file_paths:
        if os.path.exists(file_path):
            with open(file_path, 'r') as original_file:
                original_content = original_file.read()
                copy_file_path = file_path.split('.')[0] + '_copy.' + file_path.split('.')[1]
                with open(copy_file_path, 'w') as copy_file:
                    copy_file.write(original_content)
                print(f"Contents copied from {file_path} to {copy_file_path} successfully.")
        else:
            print(f"Error: File '{file_path}' not found.")

# Example usage:
import os
file_list = ['file1.txt', 'file2.txt', 'file3.txt']  # List of file paths
copy_files(file_list)
```

EXPLANATION

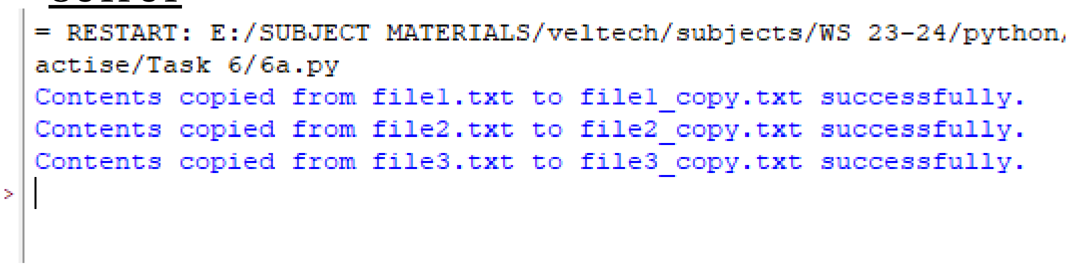


```
file_path.split('.')[0] + '_copy.' + file_path.split('.')
```

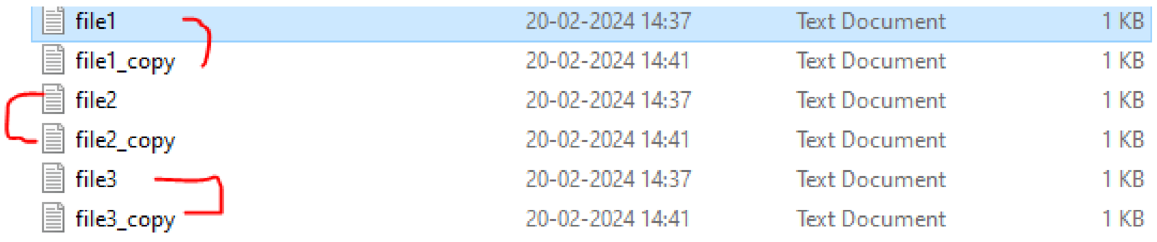
1. `file_path.split('.')`: This expression splits the file path into parts based on the dot (`'.'`) character. For example, if `file_path` is `'file1.txt'`, this expression would result in `['file1', 'txt']`.
2. `file_path.split('.')[0]`: This retrieves the first part of the split result, which represents the filename without the extension. In our example, it would be `'file1'`.
3. `'_copy.'`: This is a string literal representing the suffix that will be appended to the filename to indicate it's a copy. It includes the underscore (`_`) followed by the word "copy" and a dot (`.`).
4. `file_path.split('.')[1]`: This retrieves the second part of the split result, which represents the file extension. In our example, it would be `'txt'`.
5. Concatenation (`+`): This operator concatenates the above parts together to form the new file path. It joins the filename without extension, the "_copy." suffix, and the file extension.
   So, for our example `'file1.txt'`, the result would be `'file1_copy.txt'`, which represents the new file path for the copied file.

This approach ensures that the new file retains the original filename, with "_copy" appended before the file extension.

OUTPUT



```
= RESTART: E:/SUBJECT MATERIALS/veltech/subjects/WS 23-24/python,
actise/Task 6/6a.py
Contents copied from file1.txt to file1_copy.txt successfully.
Contents copied from file2.txt to file2_copy.txt successfully.
Contents copied from file3.txt to file3_copy.txt successfully.
>
```



```
file1         20-02-2024 14:37   Text Document   1 KB
file1_copy    20-02-2024 14:41   Text Document   1 KB
file2         20-02-2024 14:37   Text Document   1 KB
file2_copy    20-02-2024 14:41   Text Document   1 KB
file3         20-02-2024 14:37   Text Document   1 KB
file3_copy    20-02-2024 14:41   Text Document   1 KB
```

PROBLEM 2

Write a python program to create a file and display the contents (dynamic number of lines) and count the occurrence of the letter in the file and display the count.

INPUT

sample.txt

4
eLab
eLab eLab
eLab eLab tool
eLab eLab eLab eLab tool
sample.txt
e

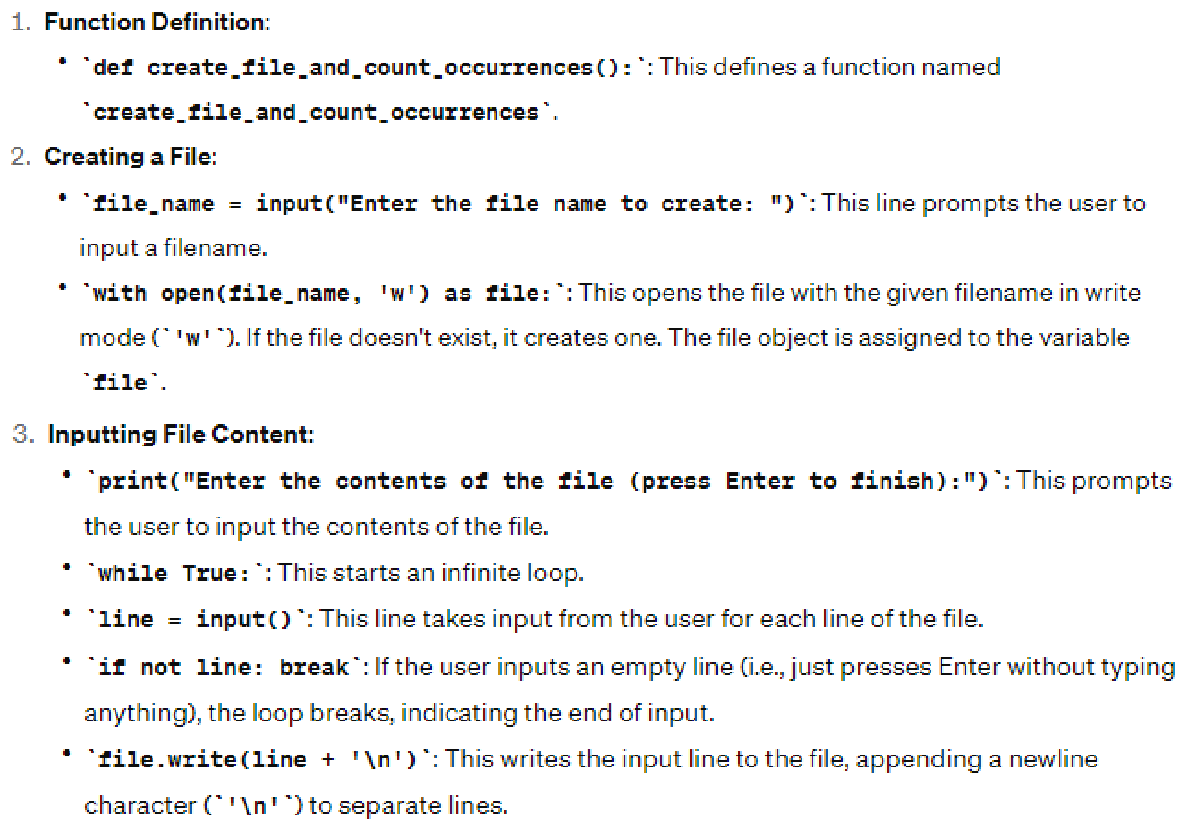
OUTPUT

Occurrences of the letter
9

PROGRAM

```
def create_file_and_count_occurrences():
    # Creating a file
    file_name = input("Enter the file name to create: ")
    with open(file_name, 'w') as file:
        print("Enter the contents of the file (press Enter to finish):")
        while True:
            line = input()
            if not line:
                break
            file.write(line + '\n')

    # Displaying contents of the file
    print("\nContents of the file:")
    with open(file_name, 'r') as file:
        print(file.read())

    # Counting occurrences of a letter
    letter = input("\nEnter the letter to count occurrences: ")
    with open(file_name, 'r') as file:
        content = file.read()
        count = content.count(letter)
        print(f"\nThe letter '{letter}' occurs {count} times in the file.")

# Calling the function
create_file_and_count_occurrences()
```

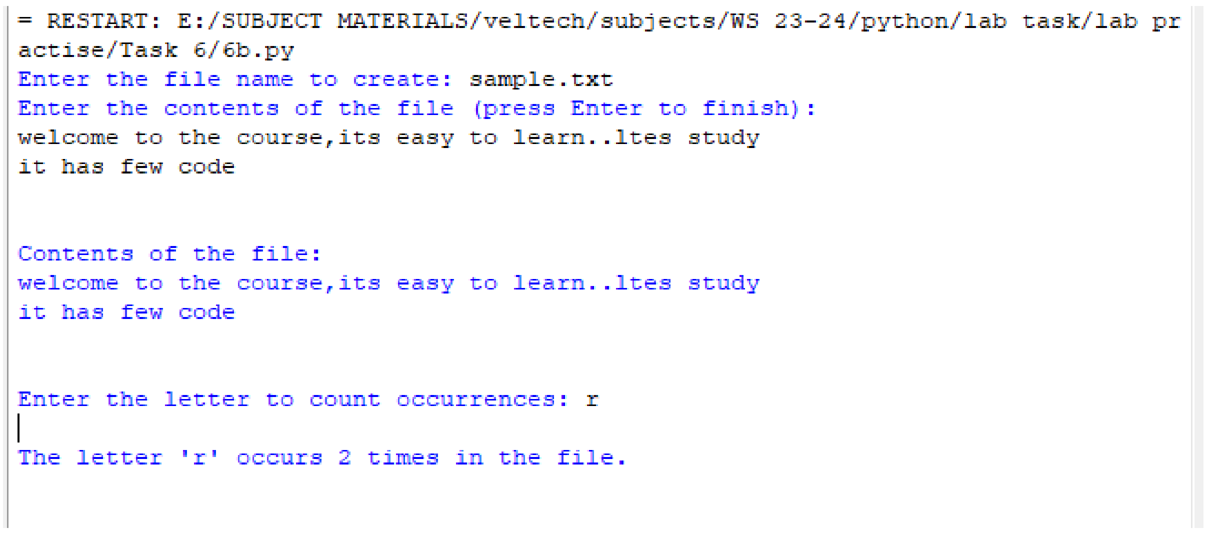
<u>EXPLANATION</u>



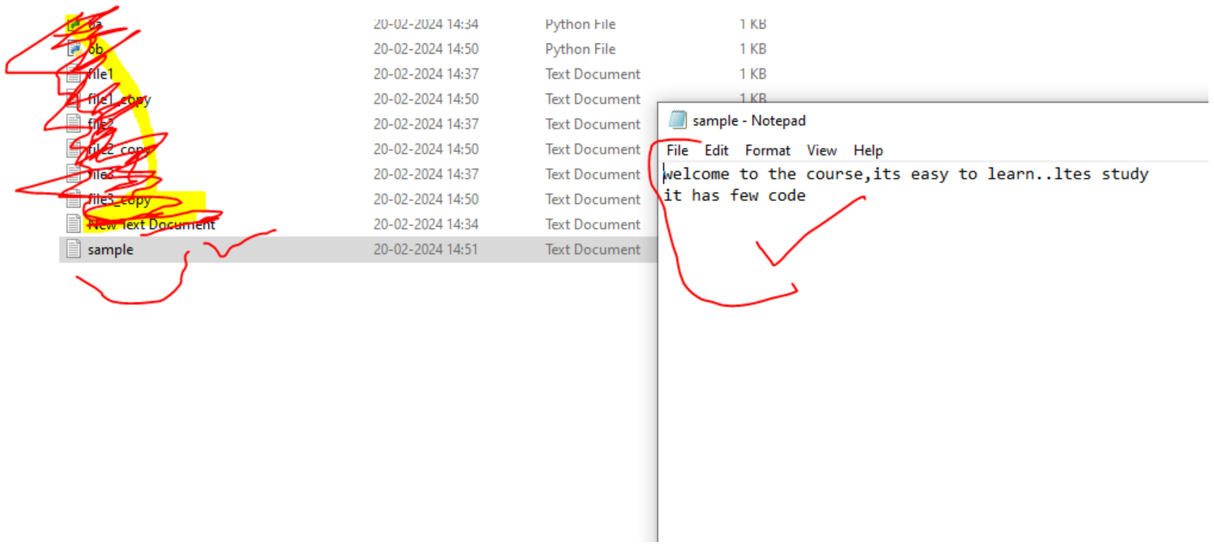
1. **Function Definition**:
   - `` `def create_file_and_count_occurrences():` ``: This defines a function named `` `create_file_and_count_occurrences` ``.
2. **Creating a File**:
   - `` `file_name = input("Enter the file name to create: ")` ``: This line prompts the user to input a filename.
   - `` `with open(file_name, 'w') as file:` ``: This opens the file with the given filename in write mode (`` `'w'` ``). If the file doesn't exist, it creates one. The file object is assigned to the variable `` `file` ``.
3. **Inputting File Content**:
   - `` `print("Enter the contents of the file (press Enter to finish):")` ``: This prompts the user to input the contents of the file.
   - `` `while True:` ``: This starts an infinite loop.
   - `` `line = input()` ``: This line takes input from the user for each line of the file.
   - `` `if not line: break` ``: If the user inputs an empty line (i.e., just presses Enter without typing anything), the loop breaks, indicating the end of input.
   - `` `file.write(line + '\n')` ``: This writes the input line to the file, appending a newline character (`` `'\n'` ``) to separate lines.

4. **Displaying File Contents:**
   - `print("\nContents of the file:")`: This line prints a header indicating that the file contents will be displayed.
   - `with open(file_name, 'r') as file:`: This opens the file again, this time in read mode (`'r'`). The file object is assigned to the variable `file`.
   - `print(file.read())`: This reads the entire contents of the file using `file.read()` and prints it to the console.
5. **Counting Occurrences of a Letter:**
   - `letter = input("\nEnter the letter to count occurrences: ")`: This prompts the user to input a letter for which they want to count occurrences.
   - `with open(file_name, 'r') as file:`: This opens the file again in read mode.
   - `content = file.read()`: This reads the entire content of the file and assigns it to the variable `content`.
   - `count = content.count(letter)`: This counts the occurrences of the specified letter in the file content using the `count()` method of strings.
   - `print(f"\nThe letter '{letter}' occurs {count} times in the file.")`: This line prints the count of occurrences of the specified letter in the file.
6. **Function Invocation:**
   - `create_file_and_count_occurrences()`: This line calls the `create_file_and_count_occurrences()` function, starting the execution of the program.

OUTPUT



```
= RESTART: E:/SUBJECT MATERIALS/veltech/subjects/WS 23-24/python/lab task/lab pr
actise/Task 6/6b.py
Enter the file name to create: sample.txt
Enter the contents of the file (press Enter to finish):
welcome to the course,its easy to learn..ltes study
it has few code


Contents of the file:
welcome to the course,its easy to learn..ltes study
it has few code


Enter the letter to count occurrences: r

The letter 'r' occurs 2 times in the file.
```



```
6a                  20-02-2024 14:34   Python File      1 KB
6b                  20-02-2024 14:50   Python File      1 KB
file1               20-02-2024 14:37   Text Document    1 KB
file1_copy          20-02-2024 14:50   Text Document    1 KB
file2               20-02-2024 14:37   Text Document
file2_copy          20-02-2024 14:50   Text Document
file3               20-02-2024 14:37   Text Document
file3_copy          20-02-2024 14:50   Text Document
New Text Document   20-02-2024 14:34   Text Document
sample              20-02-2024 14:51   Text Document
```

```
sample - Notepad
File  Edit  Format  View  Help
welcome to the course,its easy to learn..ltes study
it has few code
```

Problem 3:

Write a python program to create a file and display the contents (dynamic number of lines) and count the number of words in the file.

INPUT

sample.txt
2
eLab an auto evaluation tool in Tamilnadu
eLab will be launched in SWAYM platform soon
sample.txt

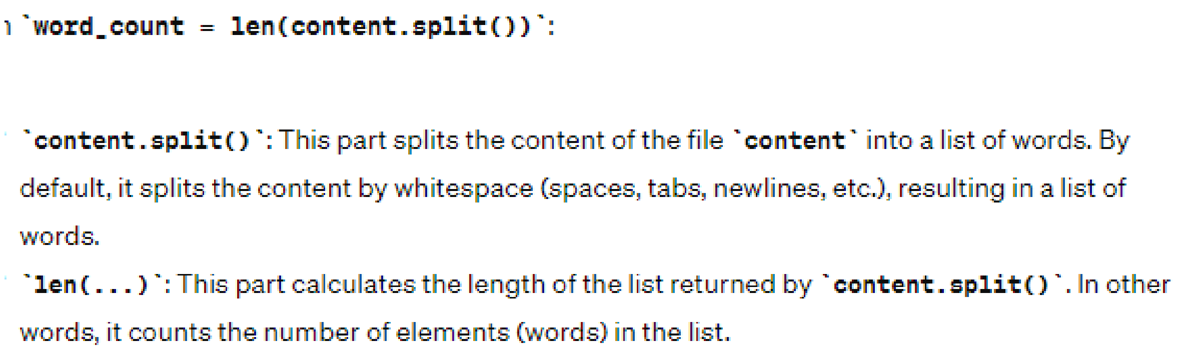
OUTPUT

Number of words:
15

PROGRAM

```
def create_file_and_count_words():

   # Creating a file

   file_name = input("Enter the file name to create: ")

   with open(file_name, 'w') as file:

      print("Enter the contents of the file (press Enter to finish):")

      while True:

         line = input()

         if not line:

            break

         file.write(line + '\n')

  # Displaying contents of the file

   print("\nContents of the file:")

   with open(file_name, 'r') as file:

      print(file.read())

   # Counting words in the file

   with open(file_name, 'r') as file:

      content = file.read()

      word_count = len(content.split())

      print(f"\nThe number of words in the file is: {word_count}")

# Calling the function

create_file_and_count_words()
```

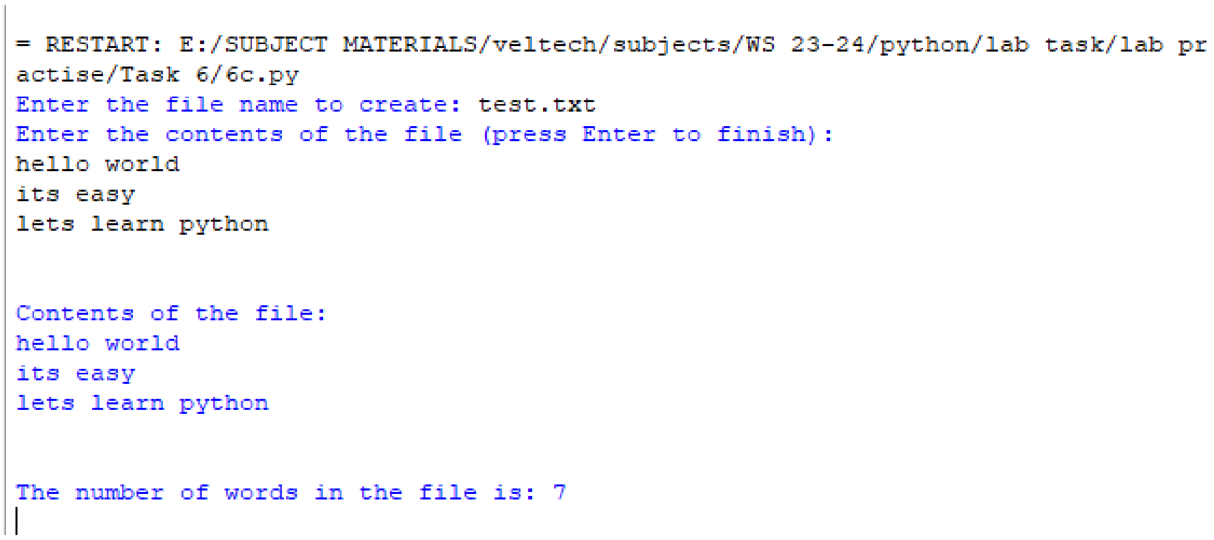
EXLANATION



1 `word_count = len(content.split())`:

- `content.split()`: This part splits the content of the file `content` into a list of words. By default, it splits the content by whitespace (spaces, tabs, newlines, etc.), resulting in a list of words.
- `len(...)`: This part calculates the length of the list returned by `content.split()`. In other words, it counts the number of elements (words) in the list.

OUTPUT



```
= RESTART: E:/SUBJECT MATERIALS/veltech/subjects/WS 23-24/python/lab task/lab pr
actise/Task 6/6c.py
Enter the file name to create: test.txt
Enter the contents of the file (press Enter to finish):
hello world
its easy
lets learn python


Contents of the file:
hello world
its easy
lets learn python


The number of words in the file is: 7
```

PROBLEM 4

You work for a government agriculture department responsible for monitoring and analyzing quality of apple produced across various regions. You've been tasked with developing a Python program to read a CSV file containing agricultural data and display its contents.

Input

include the dataset from kaggle repository

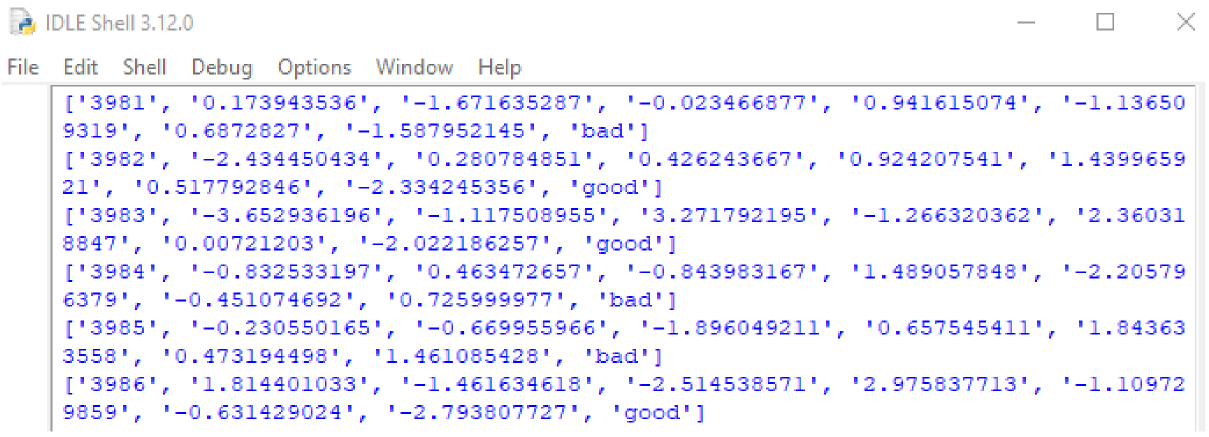
Output

Display the ontents of csv file.

PROGRAM

```
import csv

with open('apple_quality.csv','r') as file:

   data=csv.reader(file)

   for row in data:

      print(row)
```

OUTPUT



```
['3981', '0.173943536', '-1.671635287', '-0.023466877', '0.941615074', '-1.13650
9319', '0.6872827', '-1.587952145', 'bad']
['3982', '-2.434450434', '0.280784851', '0.426243667', '0.924207541', '1.4399659
21', '0.517792846', '-2.334245356', 'good']
['3983', '-3.652936196', '-1.117508955', '3.271792195', '-1.266320362', '2.36031
8847', '0.00721203', '-2.022186257', 'good']
['3984', '-0.832533197', '0.463472657', '-0.843983167', '1.489057848', '-2.20579
6379', '-0.451074692', '0.725999977', 'bad']
['3985', '-0.230550165', '-0.669955966', '-1.896049211', '0.657545411', '1.84363
3558', '0.473194498', '1.461085428', 'bad']
['3986', '1.814401033', '-1.461634618', '-2.514538571', '2.975837713', '-1.10972
9859', '-0.631429024', '-2.793807727', 'good']
```