



Instituto Tecnológico y de Estudios Superiores de Monterrey

Pruebas de software y aseguramiento de la calidad

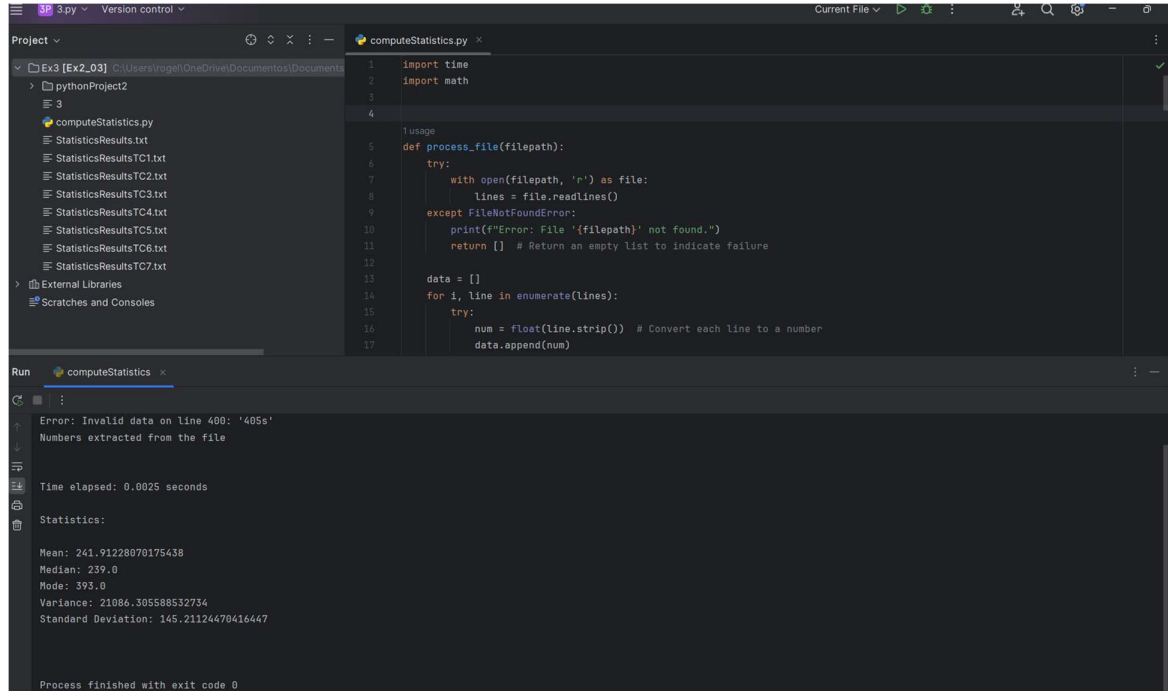
Actividad 4.2. Ejercicio de programación 1

Rogelio Abraham Rodríguez Distancia

A00989593

Resultados de P1

Los resultados se adjuntan en la carpeta del Github, renombrados de acuerdo con cada TC.



```
1 import time
2 import math
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
```

```
def process_file(filepath):
    try:
        with open(filepath, 'r') as file:
            lines = file.readlines()
    except FileNotFoundError:
        print(f"Error: File '{filepath}' not found.")
        return [] # Return an empty list to indicate failure

    data = []
    for i, line in enumerate(lines):
        try:
            num = float(line.strip()) # Convert each line to a number
            data.append(num)
```

Error: Invalid data on line 400: '405s'
Numbers extracted from the file

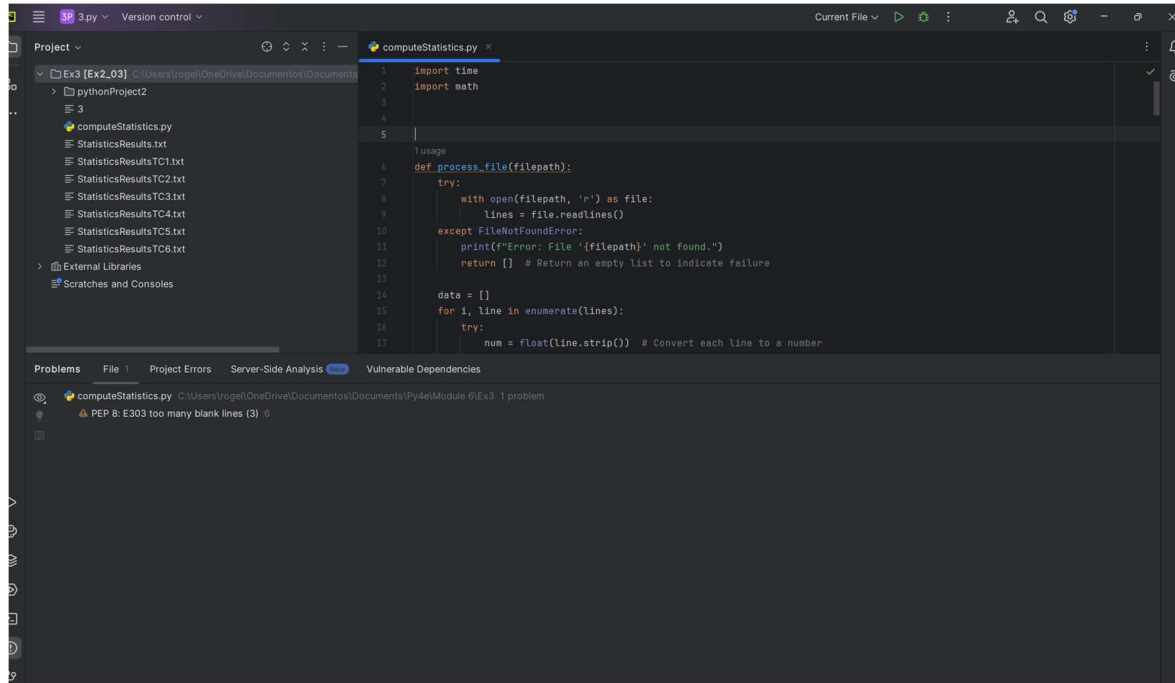
Time elapsed: 0.0025 seconds

Statistics:

Mean: 241.91228870175438
Median: 239.0
Mode: 393.0
Variance: 21086.305588532734
Standard Deviation: 145.21124470416447

Process finished with exit code 0

Evidencia con 1 error con el plug in de pycharm de plynt:



```
1 import time
2 import math
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
```

```
def process_file(filepath):
    try:
        with open(filepath, 'r') as file:
            lines = file.readlines()
    except FileNotFoundError:
        print(f"Error: File '{filepath}' not found.")
        return [] # Return an empty list to indicate failure

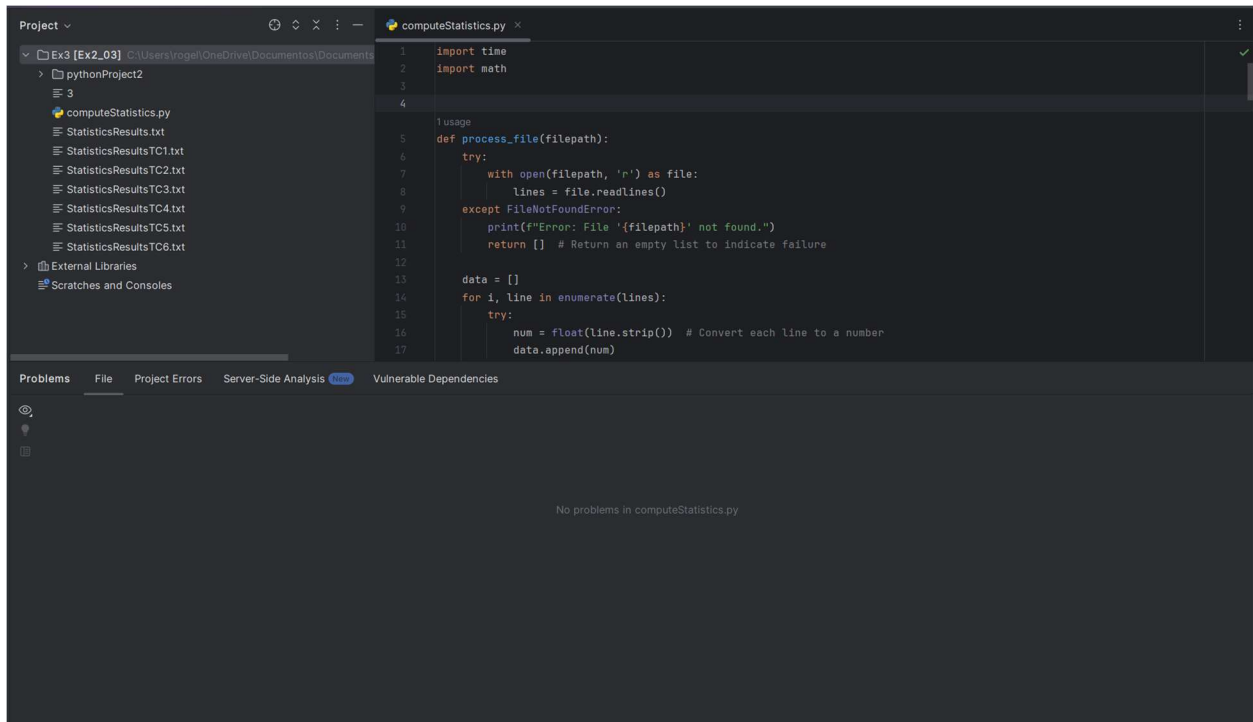
    data = []
    for i, line in enumerate(lines):
        try:
            num = float(line.strip()) # Convert each line to a number
```

Problems

computeStatistics.py: C:\Users\roge\OneDrive\Documents\Documents\Pydel\Module 6\Ex3_1 problem

PEP 8: E303 too many blank lines (3)

Evidencia sin error:

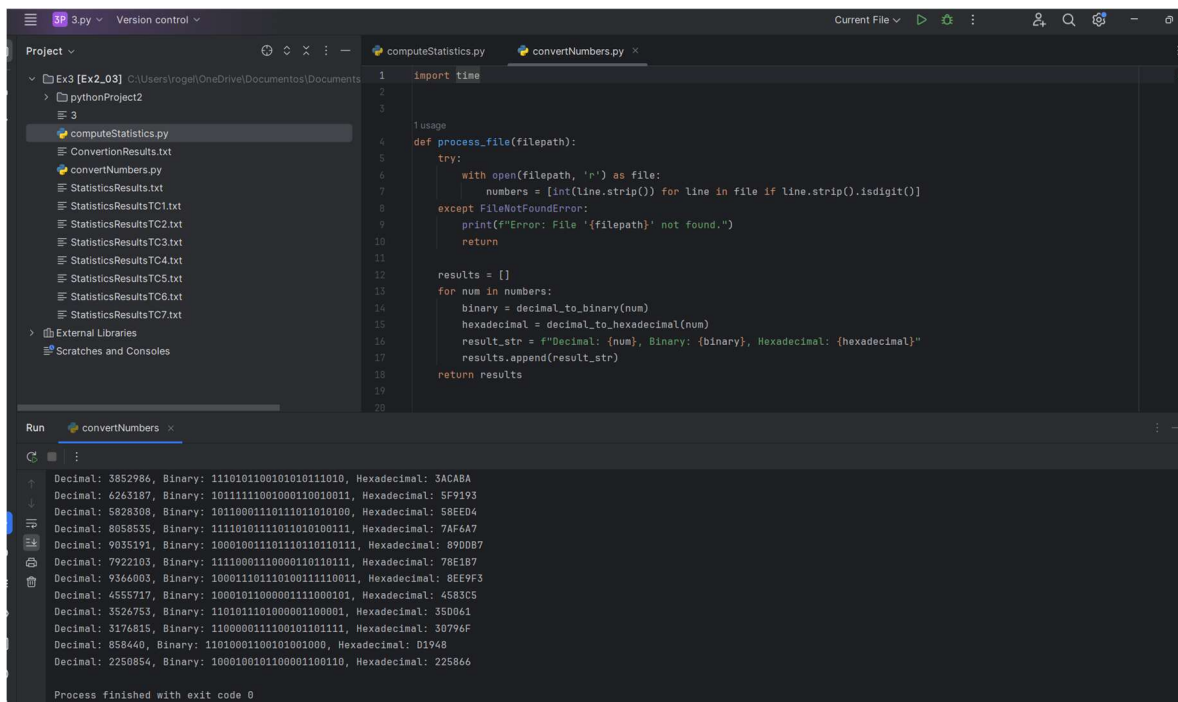


```
1 import time
2 import math
3
4
5 def process_file(filepath):
6     try:
7         with open(filepath, 'r') as file:
8             lines = file.readlines()
9         except FileNotFoundError:
10            print(f"Error: File '{filepath}' not found.")
11            return [] # Return an empty list to indicate failure
12
13    data = []
14    for i, line in enumerate(lines):
15        try:
16            num = float(line.strip()) # Convert each line to a number
17            data.append(num)
```

No problems in computeStatistics.py

Problema 2

Los resultados se adjuntan en la carpeta del Github, renombrados de acuerdo con cada TC.



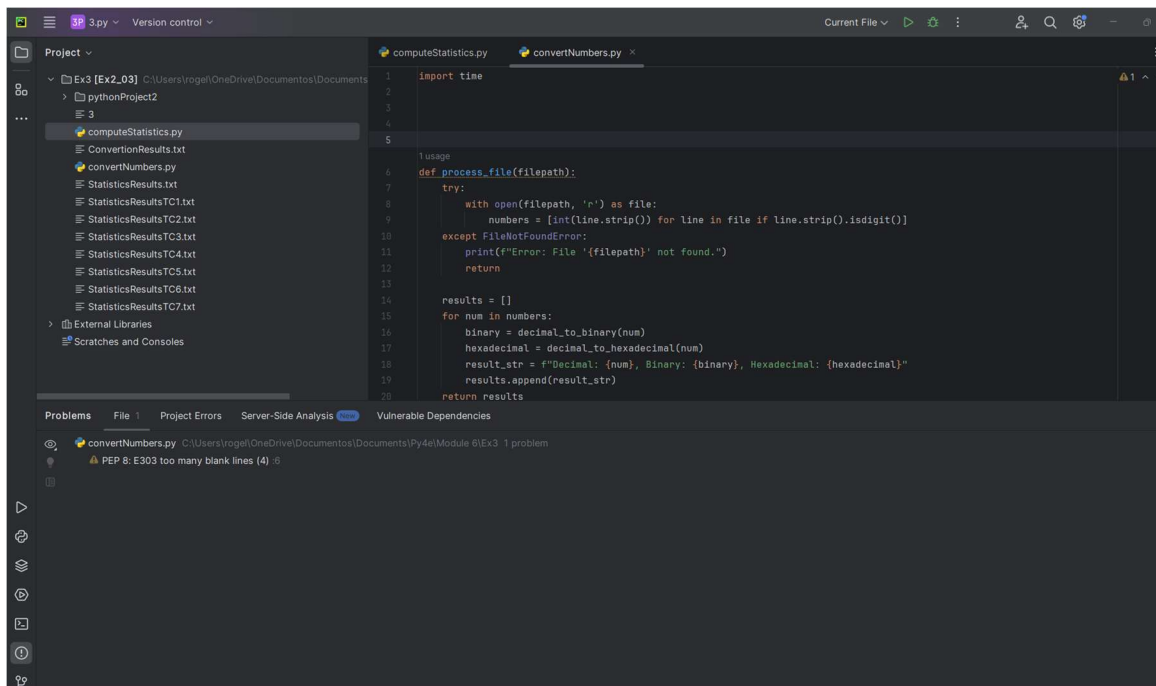
```
1 import time
2
3
4 def process_file(filepath):
5     try:
6         with open(filepath, 'r') as file:
7             numbers = [int(line.strip()) for line in file if line.strip().isdigit()]
8         except FileNotFoundError:
9             print(f"Error: File '{filepath}' not found.")
10            return
11
12    results = []
13    for num in numbers:
14        binary = decimal_to_binary(num)
15        hexadecimal = decimal_to_hexadecimal(num)
16        result_str = f"Decimal: {num}, Binary: {binary}, Hexadecimal: {hexadecimal}"
17        results.append(result_str)
18    return results
```

Run convertNumbers

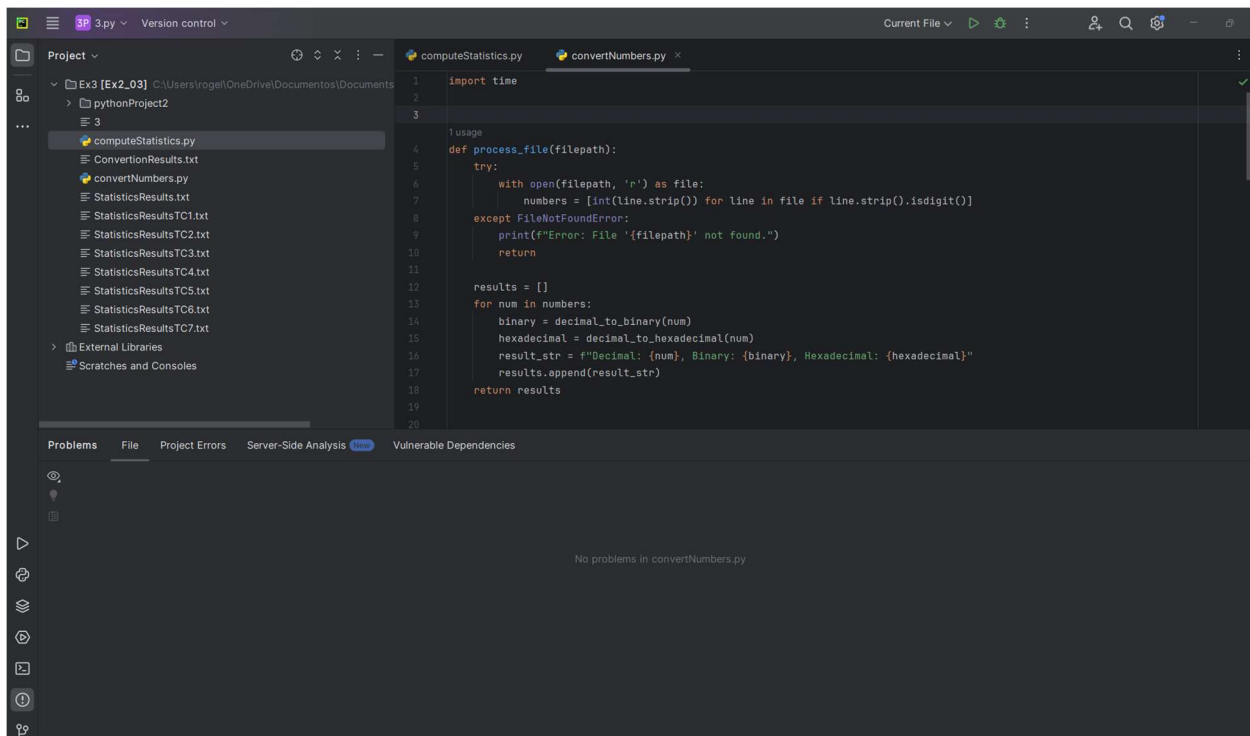
Decimal: 3852986, Binary: 1110101100101010111010, Hexadecimal: 3ACABA
Decimal: 4263187, Binary: 10111111001000110010011, Hexadecimal: 5F9193
Decimal: 5828308, Binary: 101100011101101010100, Hexadecimal: 58EE04
Decimal: 8058535, Binary: 1111010111101010100111, Hexadecimal: 7AF6A7
Decimal: 9035191, Binary: 1000100111011010101011, Hexadecimal: 890DB7
Decimal: 7922103, Binary: 11110001110000110110111, Hexadecimal: 78E1B7
Decimal: 9366003, Binary: 100011101110100111110011, Hexadecimal: 8EE9F3
Decimal: 4555717, Binary: 1000101100000111000101, Hexadecimal: 4583C5
Decimal: 3526753, Binary: 110101110100001100001, Hexadecimal: 350061
Decimal: 3176815, Binary: 11000001110010101111, Hexadecimal: 30796F
Decimal: 858440, Binary: 11010001100101001000, Hexadecimal: D1948
Decimal: 2250854, Binary: 1000100101100001100110, Hexadecimal: 225866

Process finished with exit code 0

Evidencia con 1 error con el plug in de pycharm de pylint:

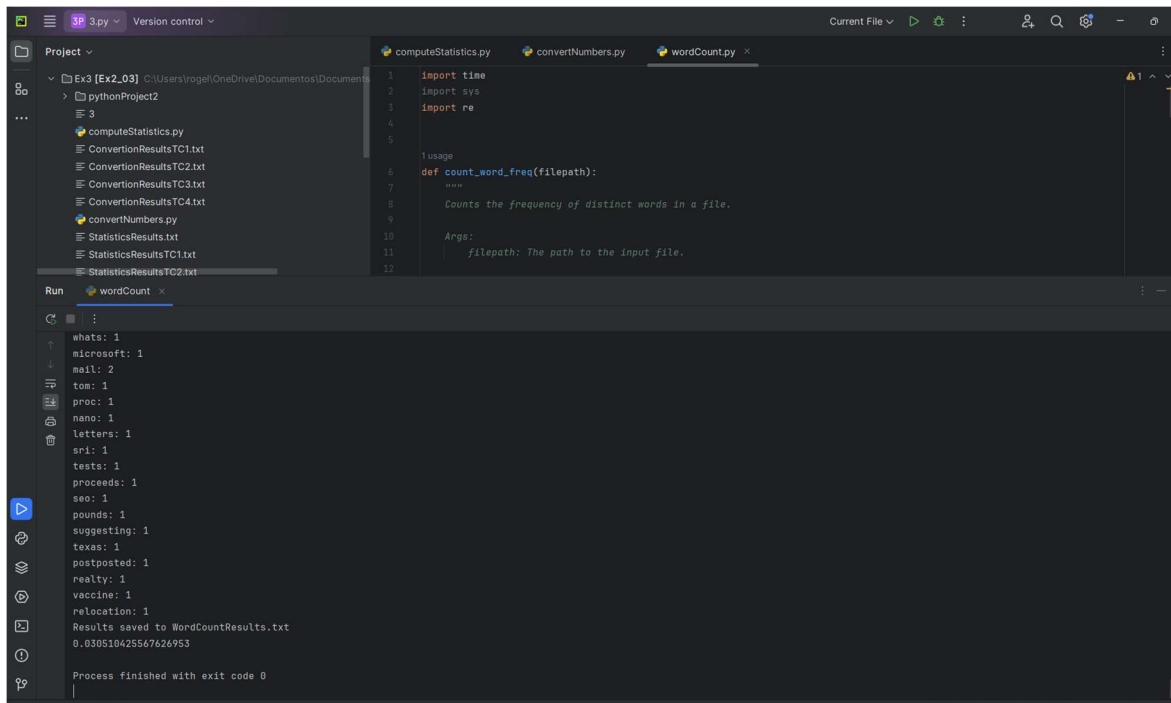


Evidencia sin error:



Problema 3

Los resultados se adjuntan en la carpeta del Github, renombrados de acuerdo con cada TC.

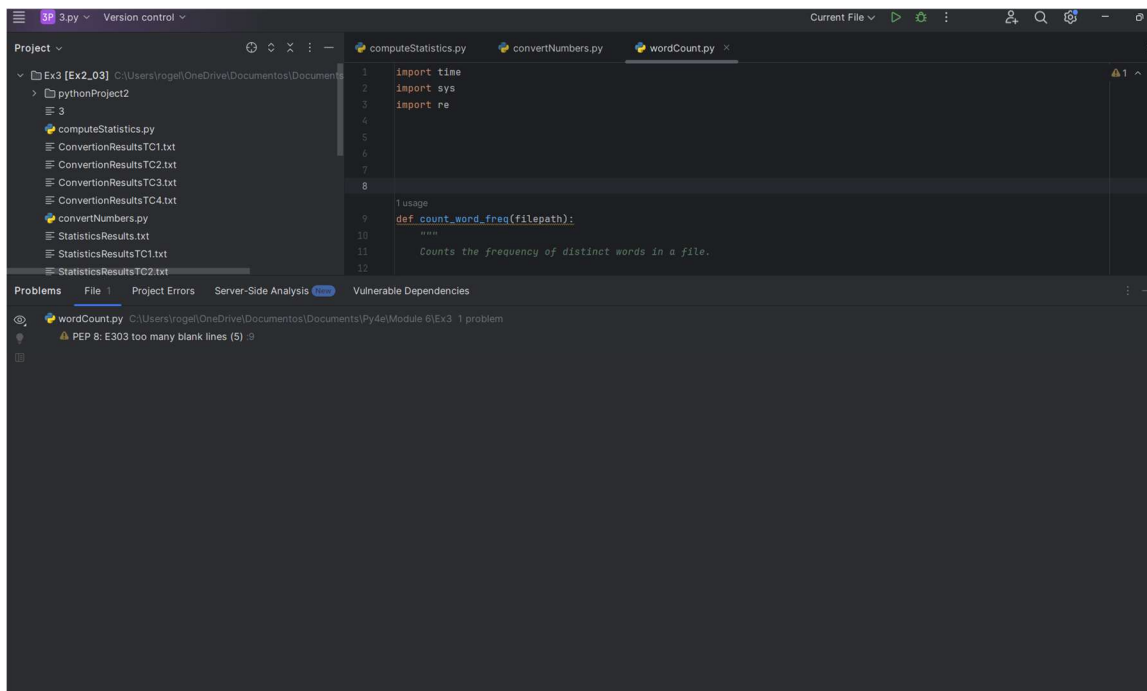


```
1 import time
2 import sys
3 import re
4
5
6 def count_word_freq(filepath):
7     """
8     Counts the frequency of distinct words in a file.
9
10    Args:
11        filepath: The path to the input file.
12
```

Run wordCount x

whats: 1
microsoft: 1
mail: 2
tom: 1
proc: 1
nano: 1
letters: 1
spi: 1
tests: 1
proceeds: 1
seo: 1
pounds: 1
suggesting: 1
texas: 1
postposted: 1
reality: 1
vaccine: 1
relocation: 1
Results saved to WordCountResults.txt
0.030510425567626953
Process finished with exit code 0

Evidencia con 1 error con el plug in de pycharm de pylint:



```
1 import time
2 import sys
3 import re
4
5
6
7
8
9
10 def count_word_freq(filepath):
11     """
12     Counts the frequency of distinct words in a file.
13
```

Problems File 1 Project Errors Server-Side Analysis Vulnerable Dependencies

wordCount.py C:\Users\roge\OneDrive\Documents\Py4e\Module 6\Ex3 1 problem

PEP 8: E303 too many blank lines (5) 9

Evidencia sin error:

