

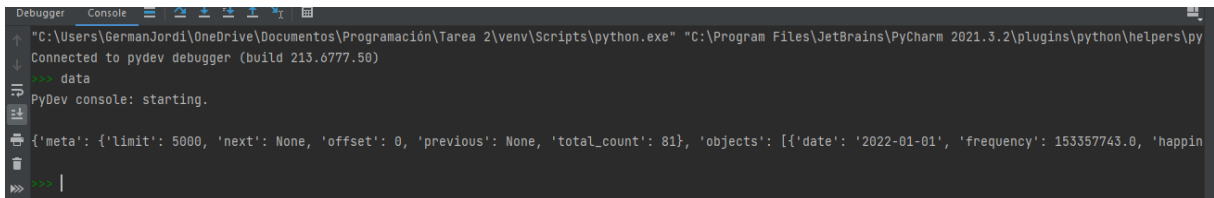
TAREA 2

Nombre: German Jordi Arreortúa Reyes

Materia: Programación avanzada.

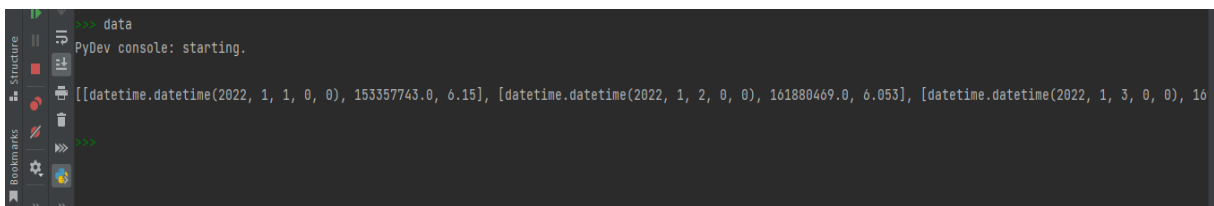
Se ejecutó el programa con el debugger en Pycharm y se obtuvo lo siguiente

La variable data cuando llega a la línea 99 tiene una estructura de diccionario como a continuación se muestra.



```
Debugger Console
"C:\Users\GermanJordi\OneDrive\Documentos\Programación\Tarea 2\venv\Scripts\python.exe" "C:\Program Files\JetBrains\PyCharm 2021.3.2\plugins\python\helpers\pydev\pydevd.py"
Connected to pydev debugger (build 213.6777.50)
>>> data
PyDev console: starting.
{'meta': {'limit': 5000, 'next': None, 'offset': 0, 'previous': None, 'total_count': 81}, 'objects': [{'date': '2022-01-01', 'frequency': 153357743.0, 'happin
```

Mientras que después de la línea 102 data tiene una estructura de lista de listas. Como a continuación se muestra.

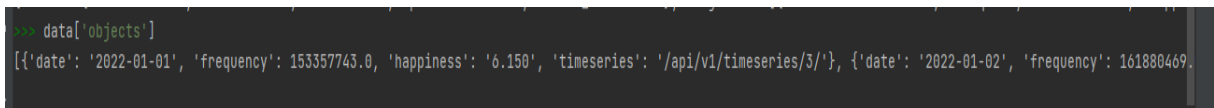


```
data
PyDev console: starting.
[[datetime.datetime(2022, 1, 1, 0, 0), 153357743.0, 6.15], [datetime.datetime(2022, 1, 2, 0, 0), 161880469.0, 6.053], [datetime.datetime(2022, 1, 3, 0, 0), 16
```

Esto se creó mediante una comprensión de lista (líneas 99 a 102).

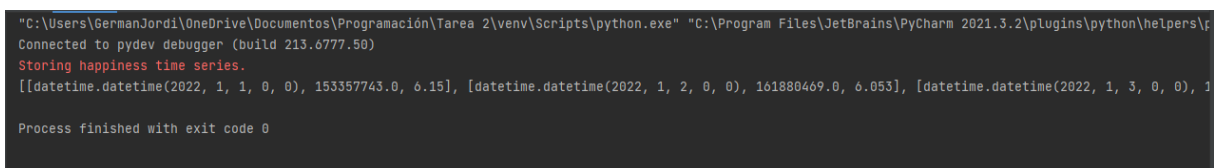
```
data = [[
    datetime.datetime.strptime(d['date'], "%Y-%m-%d"), d['frequency'],
    float(d['happiness'])
] for d in data['objects']]
```

En la comprensión de lista el bucle for corre sobre todos los elementos de data['objects'] la cuál es una lista de diccionarios y se hace la asignación del tipo de dato correspondiente en datetime.datetime.strptime().



```
data['objects']
[{'date': '2022-01-01', 'frequency': 153357743.0, 'happiness': '6.150', 'timeseries': '/api/v1/timeseries/3/'}, {'date': '2022-01-02', 'frequency': 161880469.
```

Ahora la diferencia entre usar print(data) y print('\n'.join(str(d) for d in data)) en la línea 116 es que print(data) imprime la variable data como una lista, pues data es de tipo lista después de la línea 102.



```
"C:\Users\GermanJordi\OneDrive\Documentos\Programación\Tarea 2\venv\Scripts\python.exe" "C:\Program Files\JetBrains\PyCharm 2021.3.2\plugins\python\helpers\pydev\pydevd.py"
Connected to pydev debugger (build 213.6777.50)
Storing happiness time series.
[[datetime.datetime(2022, 1, 1, 0, 0), 153357743.0, 6.15], [datetime.datetime(2022, 1, 2, 0, 0), 161880469.0, 6.053], [datetime.datetime(2022, 1, 3, 0, 0), 161880469.0, 6.053]]
Process finished with exit code 0
```

Mientras que `print("\n".join(str(d) for d in data))` utiliza `join` el cual une los elementos de un objeto iterable en una cadena mediante una cadena como separador.

En `print('\n'.join(str(d) for d in data))` mediante el `for` se transforma cada elemento de la lista `date` en cadena, luego con `'\n'.join` une cada cadena por medio de `'\n'`. Así pues `print('\n'.join(str(d) for d in data))` imprime cadenas a diferencia de `print(data)` que imprime una lista.

```
C:\Users\GermanJordi\OneDrive\Documents\Programaci6n\Tarea 2\venv\Scripts\python.exe "C:\Program Files\JetBrains\PyCharm 2021.3.2\plugins\python\helpers\pydev\_pydevd_bundle\pydevd.py" -x --no-coverage --multiprocess Connected to pydev debugger (build 213.6777.50) Storing happiness time series.
```

Date	Happiness
[datetime.datetime(2022, 1, 1, 0, 0), 153357743.0, 6.15]	
[datetime.datetime(2022, 1, 2, 0, 0), 161880469.0, 6.053]	
[datetime.datetime(2022, 1, 3, 0, 0), 169773375.0, 6.034]	
[datetime.datetime(2022, 1, 4, 0, 0), 172208177.0, 6.014]	
[datetime.datetime(2022, 1, 5, 0, 0), 181196294.0, 5.987]	
[datetime.datetime(2022, 1, 6, 0, 0), 180295617.0, 5.988]	
[datetime.datetime(2022, 1, 7, 0, 0), 177795631.0, 6.007]	