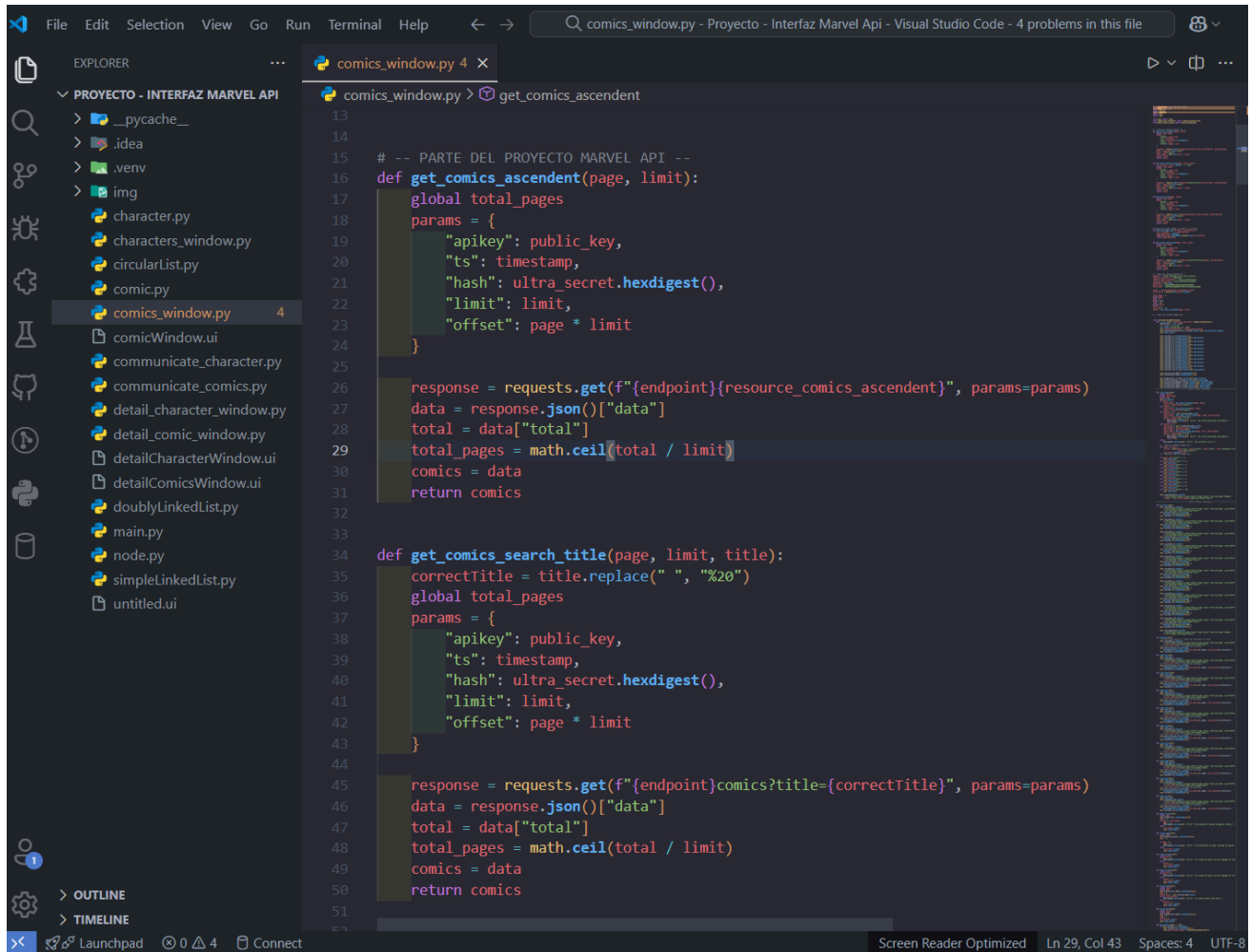


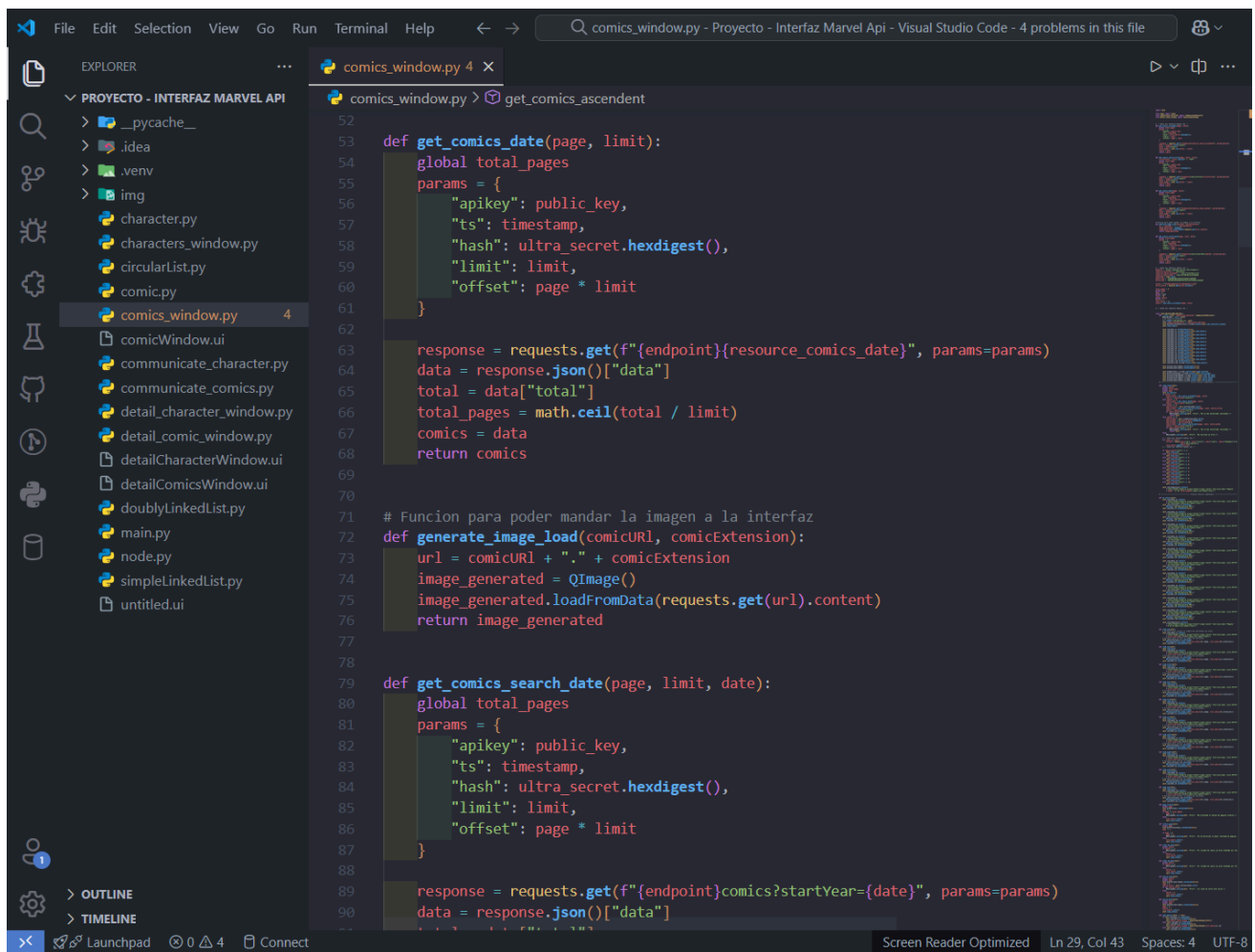
Código Incorrecto a Código Limpio

Código Estructura de Datos I

1. Funciones desordenadas (No colocadas en orden de invocación)



```
13
14
15 # -- PARTE DEL PROYECTO MARVEL API --
16 def get_comics_ascendent(page, limit):
17     global total_pages
18     params = {
19         "apikey": public_key,
20         "ts": timestamp,
21         "hash": ultra_secret.hexdigest(),
22         "limit": limit,
23         "offset": page * limit
24     }
25
26     response = requests.get(f"{endpoint}{resource_comics_ascendent}", params=params)
27     data = response.json()["data"]
28     total = data["total"]
29     total_pages = math.ceil(total / limit)
30     comics = data
31     return comics
32
33
34 def get_comics_search_title(page, limit, title):
35     correctTitle = title.replace(" ", "%20")
36     global total_pages
37     params = {
38         "apikey": public_key,
39         "ts": timestamp,
40         "hash": ultra_secret.hexdigest(),
41         "limit": limit,
42         "offset": page * limit
43     }
44
45     response = requests.get(f"{endpoint}comics?title={correctTitle}", params=params)
46     data = response.json()["data"]
47     total = data["total"]
48     total_pages = math.ceil(total / limit)
49     comics = data
50     return comics
51
```



la función **get_comics_search_title** está declarada antes de la función **get_comics_date** y la función **get_comics_date** está siendo invocada en la línea 173 mientras que la función **get_comics_search_title** está siendo invocada en la línea 177

```

176         search_title = self.textComicName.text()
177         comics_data = get_comics_search_title(page, limit, search_title)
178         comics = comics_data["results"]
179         if comics_data["count"] == 0:

```

*Invocación de la función **get_comics_search_title***

```

172         elif option == 2:
173             comics_data = get_comics_date(page, limit)
174             comics = comics_data["results"]

```

*Invocación de la función **get_comics_date***

De igual manera con las demás funciones declaradas al inicio

```

310         QPixmap(generate_image_load(list_comics[0].image, list_comics[0].extension)))
311         self.comicBtn_c1.setEnabled(True)

```

Generate_image_load

```

186         comics_data = get_comics_search_date(page, limit, search_date)
187         comics = comics_data["results"]

```

get_comics_search_cate

2. Variables declaradas con nombres confusos

```
106 access = { "timestamp": timestamp, "secret_key": secret_key, "public_key": public_key }
107 ultra_secret = hashlib.md5(access.encode())
108
```

```
20     "ts": timestamp,
21     "hash": ultra_secret.hexdigest(),
22     "limit": limit,
```

ultra_secret es una variable que se confundo

```
    "ts": timestamp,
    "hash": auth_hash.hexdigest(),
    "limit": limit,
```

```
106 access = { "timestamp": timestamp, "secret_key": secret_key, "public_key": public_key }
107 auth_hash = hashlib.md5(access.encode())
108
```

Se cambio para auth_hash

3. Comentario sin sentido

```
# -- PARTE DEL PROYECTO MARVEL API --
```

Este comentario no aporta nada al proyecto

```
116 list_comics = []
117 comics = get_comics_ascendent(page, limit)
118
119
120 class ComicsWindow(QMainWindow):
121     def __init__(self, communicate_character: CommunicateCharacter):
122         QMainWindow.__init__(self)
123         # Se carga la interfaz
124         uic.loadUi("comicWindow.ui", (parameter) communicate_character)
```

Comentario eliminado