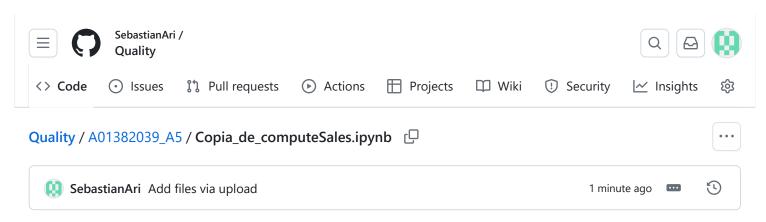
Sebastián Arizpe Sáenz

A01382039

Github Permalink:

 $\frac{\text{https://github.com/SebastianAri/Quality/tree/47754954a5940fe402df57caa92502cbd393769}}{2/\text{A}01382039_\text{A}5}$



185 lines (185 loc) · 5.53 KB

```
In [1]: import json import time
```

```
↑ Тор
Quality / A01382039_A5 / Copia_de_computeSales.ipynb
                                                                                        Raw 🖵 🕹
Preview
           Code
                    Blame
                 Load JSUN data from a file.
                 try:
                     with open(filename, 'r', encoding='utf-8') as file:
                         data = json.load(file)
                     return data
                 except FileNotFoundError:
                     print(f"Error: File '{filename}' not found.")
                     return {}
                 except json.JSONDecodeError:
                     print(f"Error: Invalid JSON format in '{filename}'.")
             def similar(a, b):
                 Calculate the similarity ratio between two strings.
                 return SequenceMatcher(None, a, b).ratio()
             def compute_total_cost(price_catalogue, sales_record):
                 Compute the total cost of sales.
                 total_cost = 0
                 for sale in sales record:
                     product name = sale["Product"]
                     match found = False
                     for catalog_product in price_catalogue:
                         similarity = similar(product_name, catalog_product["title"])
                         if similarity >= 0.9:
                             price = catalog_product["price"]
                             quantity = sale["Quantity"]
                             total_cost += price * quantity
                             match found = True
                             break
                     if not match found:
                         print(f"Warning: No match found for product '\
                         {product_name}' in price catalogue.")
                 return total cost
             def main(sales_record_file, price_catalogue_file='TC1.ProductList.json'):
                 Main function to compute total sales cost.
                 start_time = time.time()
                 price_catalogue = load_json_file(price_catalogue_file)
                 sales_record = load_json_file(sales_record_file)
```

total cost = compute total cost(price catalogue, sales record)

end_time = time.time()

```
elapsed_time = end_time - start_time
             print(f"Total cost of sales: ${total_cost:.2f}")
             print(f"Time elapsed: {elapsed_time:.2f} seconds")
             with open("SalesResults.txt", "w", encoding='utf-8') as results_file:
                 results_file.write(f"Total cost of sales: ${total_cost:.2f}\n")
                 results_file.write(f"Time elapsed: {elapsed_time:.2f} seconds\n")
In [7]:
         main('TC1.Sales.json')
       Total cost of sales: $2481.86
       Time elapsed: 0.05 seconds
       Total cost of sales: $166568.23
       Time elapsed: 0.04 seconds
In [3]:
         main('TC2.Sales.json')
       Total cost of sales: $166568.23
       Time elapsed: 0.09 seconds
In [4]: | main('TC3.Sales.json')
       Warning: No match found for product '
                                                       Elotes' in price catalogue.
       Warning: No match found for product '
                                                        Frijoles' in price catalogue.
       Total cost of sales: $165235.37
       Time elapsed: 0.10 seconds
In [5]:
         # pylint: disable = C0304
```

