

Python programming and practice

Expenditure Management Program Production Project

Progress Report #2

Date : 2023.12.08

Name : Kim Liha

ID : 204201

1. Introduction

1) Background

People are having a hard time saving due to increased consumption that they don't need due to incorrect consumption habits. You need a program that can simply check your consumption and help you develop the right consumption habits.

2) Project goal

Store the user's expenditure details by content and allow you to check the average consumption per month.

3) Differences from existing programs

If you look at the amount of consumption only in units of numbers, you may not feel it, so you can feel more intuitively how much you saved through the price of the product (e.g., coffee, chicken).

2. Functional Requirement

1) Function 1 Expenditure storage function

- ability to store one's spending, calculate the amount of spending for the month.

2) Function 2 User convenience features

- Features that increase convenience and intuition, such as storing fixed monthly expenses. (cell phone bills, utility bills, etc.)

3. Progress

1) 기능 구현

(1) Save Expenditure History

- Enter: Expenditure (spending, amount, date, category)
- Output: Saved
- Explanation: When the user enters the expenditure details, it is saved with the date in the csv file.
- Use csv, function, pandas, numpy, shutil
- code screenshot

```

1 import datetime
2 import csv
3 import pandas as pd
4 import shutil
5 from pandas import Timestamp
6 import re, os
7 import numpy as np
8
9 # matplotlib 한글 폰트
10 LabelStyle.MISC_CMAPS_MODE = False
11
12
13 # csv 파일 읽기
14
15 # 1. CSV 파일 읽기 (20221001 ~), 각 행 번호를 출력함
16 # 2. 각 행 출력 (4 행만 출력)
17 # 3. 출력 종료
18
19
20 # csv 파일 읽기
21 def save_to_csv(response_name, amount, date, category):
22     # csv 파일에 데이터를 기록함
23     data = [{"response_name": str(response_name), "date": date, "category": category}]
24
25     # CSV 파일에 데이터 쓰기
26     with open('data.csv', 'a', newline='', encoding='utf-8') as file:
27         writer = csv.writer(file)
28         writer.writerow(data)
29
30
31 # 데이터 출력, 저장 함수
32 def enter_expense():
33     # 1. 데이터 입력 (response_name, amount, date, category)
34     response_name = input("입력할 항목명 입력하세요: ")
35     amount = float(input("금액을 입력하세요: "))
36     date = input("입력할 날짜를 입력하세요 (YYYY-MM-DD 형식): ")
37     category = input("입력할 카테고리 입력하세요: ")
38
39     # 데이터 저장
40     save_to_csv(response_name, amount, date, category)
41     print("데이터가 성공적으로 저장되었습니다.")
42
43     # 데이터 출력 (data.csv)
44     shutil.copy('data.csv', 'data_backup.csv')
45
46
47 # CSV 파일 읽기
48 # 1. CSV 파일 읽기 (20221001 ~), CSV 파일 20221001부터 읽기 (41)
49
50 def csv_expenses_month():
51     df = pd.read_csv('data_backup.csv')
52     month = int(input("조회할 월을 입력하세요 (월: 1 ~ 12): "))
53
54     # 월별 총액, 평균, 최대, 최소
55     failed_month = df.loc[df['date'] == month]
56
57     # 1. 월별 총액, 평균, 최대, 최소 출력 (월: 1 ~ 12)
58     filtered_index = failed_index[failed_index['date'] == month]
59     print(LabelStyle(failed_index, headers='keys', labelstyle='grid', showindex=False, stralign='center'))
60
61
62 # 데이터 출력, 저장 함수
63 def data_conversion():
64     df = pd.read_csv('data_backup.csv')
65
66     # 1. 데이터 출력 (response_name, amount, date, category)
67     df['date'] = df['date'].str[:10]
68     df['date'] = df['date'].str[:10]
69     df['date'] = df['date'].str[:10]
70     df['date'] = df['date'].str[:10]
71     df['date'] = df['date'].str[:10]
72     df['date'] = df['date'].str[:10]
73     df['date'] = df['date'].str[:10]
74     df['date'] = df['date'].str[:10]
75     df['date'] = df['date'].str[:10]
76     df['date'] = df['date'].str[:10]
77     df['date'] = df['date'].str[:10]
78     df['date'] = df['date'].str[:10]
79     df['date'] = df['date'].str[:10]
80     df['date'] = df['date'].str[:10]
81     df['date'] = df['date'].str[:10]
82     df['date'] = df['date'].str[:10]
83     df['date'] = df['date'].str[:10]
84     df['date'] = df['date'].str[:10]
85     df['date'] = df['date'].str[:10]
86     df['date'] = df['date'].str[:10]
87     df['date'] = df['date'].str[:10]
88     df['date'] = df['date'].str[:10]
89     df['date'] = df['date'].str[:10]
90     df['date'] = df['date'].str[:10]
91     df['date'] = df['date'].str[:10]
92     df['date'] = df['date'].str[:10]
93     df['date'] = df['date'].str[:10]
94     df['date'] = df['date'].str[:10]
95     df['date'] = df['date'].str[:10]
96     df['date'] = df['date'].str[:10]
97     df['date'] = df['date'].str[:10]
98     df['date'] = df['date'].str[:10]
99     df['date'] = df['date'].str[:10]
100     df['date'] = df['date'].str[:10]
101     df['date'] = df['date'].str[:10]
102     df['date'] = df['date'].str[:10]
103     df['date'] = df['date'].str[:10]
104     df['date'] = df['date'].str[:10]
105     df['date'] = df['date'].str[:10]
106     df['date'] = df['date'].str[:10]
107     df['date'] = df['date'].str[:10]
108     df['date'] = df['date'].str[:10]
109     df['date'] = df['date'].str[:10]
110     df['date'] = df['date'].str[:10]
111     df['date'] = df['date'].str[:10]
112     df['date'] = df['date'].str[:10]
113     df['date'] = df['date'].str[:10]
114     df['date'] = df['date'].str[:10]
115     df['date'] = df['date'].str[:10]
116     df['date'] = df['date'].str[:10]
117     df['date'] = df['date'].str[:10]
118     df['date'] = df['date'].str[:10]
119     df['date'] = df['date'].str[:10]
120     df['date'] = df['date'].str[:10]
121     df['date'] = df['date'].str[:10]
122     df['date'] = df['date'].str[:10]
123     df['date'] = df['date'].str[:10]
124     df['date'] = df['date'].str[:10]
125     df['date'] = df['date'].str[:10]
126     df['date'] = df['date'].str[:10]
127     df['date'] = df['date'].str[:10]
128     df['date'] = df['date'].str[:10]
129     df['date'] = df['date'].str[:10]
130     df['date'] = df['date'].str[:10]
131     df['date'] = df['date'].str[:10]
132     df['date'] = df['date'].str[:10]
133     df['date'] = df['date'].str[:10]
134     df['date'] = df['date'].str[:10]
135     df['date'] = df['date'].str[:10]
136     df['date'] = df['date'].str[:10]
137     df['date'] = df['date'].str[:10]
138     df['date'] = df['date'].str[:10]
139     df['date'] = df['date'].str[:10]
140     df['date'] = df['date'].str[:10]
141     df['date'] = df['date'].str[:10]
142     df['date'] = df['date'].str[:10]
143     df['date'] = df['date'].str[:10]
144     df['date'] = df['date'].str[:10]
145     df['date'] = df['date'].str[:10]
146     df['date'] = df['date'].str[:10]
147     df['date'] = df['date'].str[:10]
148     df['date'] = df['date'].str[:10]
149     df['date'] = df['date'].str[:10]
150     df['date'] = df['date'].str[:10]
151     df['date'] = df['date'].str[:10]
152     df['date'] = df['date'].str[:10]
153     df['date'] = df['date'].str[:10]
154     df['date'] = df['date'].str[:10]
155     df['date'] = df['date'].str[:10]
156     df['date'] = df['date'].str[:10]
157     df['date'] = df['date'].str[:10]
158     df['date'] = df['date'].str[:10]
159     df['date'] = df['date'].str[:10]
160     df['date'] = df['date'].str[:10]
161     df['date'] = df['date'].str[:10]
162     df['date'] = df['date'].str[:10]
163     df['date'] = df['date'].str[:10]
164     df['date'] = df['date'].str[:10]
165     df['date'] = df['date'].str[:10]
166     df['date'] = df['date'].str[:10]
167     df['date'] = df['date'].str[:10]
168     df['date'] = df['date'].str[:10]
169     df['date'] = df['date'].str[:10]
170     df['date'] = df['date'].str[:10]
171     df['date'] = df['date'].str[:10]
172     df['date'] = df['date'].str[:10]
173     df['date'] = df['date'].str[:10]
174     df['date'] = df['date'].str[:10]
175     df['date'] = df['date'].str[:10]
176     df['date'] = df['date'].str[:10]
177     df['date'] = df['date'].str[:10]
178     df['date'] = df['date'].str[:10]
179     df['date'] = df['date'].str[:10]
180     df['date'] = df['date'].str[:10]
181     df['date'] = df['date'].str[:10]
182     df['date'] = df['date'].str[:10]
183     df['date'] = df['date'].str[:10]
184     df['date'] = df['date'].str[:10]
185     df['date'] = df['date'].str[:10]
186     df['date'] = df['date'].str[:10]
187     df['date'] = df['date'].str[:10]
188     df['date'] = df['date'].str[:10]
189     df['date'] = df['date'].str[:10]
190     df['date'] = df['date'].str[:10]
191     df['date'] = df['date'].str[:10]
192     df['date'] = df['date'].str[:10]
193     df['date'] = df['date'].str[:10]
194     df['date'] = df['date'].str[:10]
195     df['date'] = df['date'].str[:10]
196     df['date'] = df['date'].str[:10]
197     df['date'] = df['date'].str[:10]
198     df['date'] = df['date'].str[:10]
199     df['date'] = df['date'].str[:10]
200     df['date'] = df['date'].str[:10]
201     df['date'] = df['date'].str[:10]
202     df['date'] = df['date'].str[:10]
203     df['date'] = df['date'].str[:10]
204     df['date'] = df['date'].str[:10]
205     df['date'] = df['date'].str[:10]
206     df['date'] = df['date'].str[:10]
207     df['date'] = df['date'].str[:10]
208     df['date'] = df['date'].str[:10]
209     df['date'] = df['date'].str[:10]
210     df['date'] = df['date'].str[:10]
211     df['date'] = df['date'].str[:10]
212     df['date'] = df['date'].str[:10]
213     df['date'] = df['date'].str[:10]
214     df['date'] = df['date'].str[:10]
215     df['date'] = df['date'].str[:10]
216     df['date'] = df['date'].str[:10]
217     df['date'] = df['date'].str[:10]
218     df['date'] = df['date'].str[:10]
219     df['date'] = df['date'].str[:10]
220     df['date'] = df['date'].str[:10]
221     df['date'] = df['date'].str[:10]
222     df['date'] = df['date'].str[:10]
223     df['date'] = df['date'].str[:10]
224     df['date'] = df['date'].str[:10]
225     df['date'] = df['date'].str[:10]
226     df['date'] = df['date'].str[:10]
227     df['date'] = df['date'].str[:10]
228     df['date'] = df['date'].str[:10]
229     df['date'] = df['date'].str[:10]
230     df['date'] = df['date'].str[:10]
231     df['date'] = df['date'].str[:10]
232     df['date'] = df['date'].str[:10]
233     df['date'] = df['date'].str[:10]
234     df['date'] = df['date'].str[:10]
235     df['date'] = df['date'].str[:10]
236     df['date'] = df['date'].str[:10]
237     df['date'] = df['date'].str[:10]
238     df['date'] = df['date'].str[:10]
239     df['date'] = df['date'].str[:10]
240     df['date'] = df['date'].str[:10]
241     df['date'] = df['date'].str[:10]
242     df['date'] = df['date'].str[:10]
243     df['date'] = df['date'].str[:10]
244     df['date'] = df['date'].str[:10]
245     df['date'] = df['date'].str[:10]
246     df['date'] = df['date'].str[:1
```

2) 테스트 결과

(1) Save Expenditure History

- Enter the expenditure history information from the user and save it to the csv file

```
::::::::::::::::::
지출 관리 프로그램
::::::::::::::::::
:::::::::메뉴:::::::::
1. 지출 내용 입력
2. 지난 지출 내역 확인
3. 지출 평균 확인
4. 종료
번호를 입력해주세요: 1
지출 항목을 입력하세요: 저녁밥
금액을 입력하세요: 9500
날짜를 입력하세요 (YYYY-MM-DD 형식): 20231208
카테고리를 입력하세요: 식비
지출 정보가 저장되었습니다.
```

```
커피구매,7000.0,2023-06-05,식비,2023-06-05,2023,6,5,Monday
화장품구매,45000.0,2023-06-10,미용,2023-06-10,2023,6,10,Saturday
전기요금,55000.0,2023-06-15,생활용품,2023-06-15,2023,6,15,Thursday
운동화구매,85000.0,2023-06-20,쇼핑,2023-06-20,2023,6,20,Tuesday
의료진단비,70000.0,2023-06-25,의료/건강,2023-06-25,2023,6,25,Sunday
식료품,52000.0,2023-06-30,식비,2023-06-30,2023,6,30,Friday
여행경비,190000.0,2023-07-05,여행,2023-07-05,2023,7,5,Wednesday
음악강습,55000.0,2023-07-10,취미/교육,2023-07-10,2023,7,10,Monday
의류구매,80000.0,2023-07-15,쇼핑,2023-07-15,2023,7,15,Saturday
전화요금,35000.0,2023-07-20,통신비,2023-07-20,2023,7,20,Thursday
자동차 주유,90000.0,2023-07-25,교통비,2023-07-25,2023,7,25,Tuesday
도서구입,38000.0,2023-07-30,문화,2023-07-30,2023,7,30,Sunday
의료비,50000.0,2023-08-05,의료/건강,2023-08-05,2023,8,5,Saturday
식당외식,42000.0,2023-08-10,식비,2023-08-10,2023,8,10,Thursday
점심,8000.0,2023-12-08,식비,2023-12-08,2023,12,8,Friday
저녁밥,9500.0,2023-12-08,식비,2023-12-08,2023,12,8,Friday
```

(2) Checking spending history

- If you enter a specific month, output the expenditure history for that month

지출 내역을 조회할 달을 입력해주세요.
예시) 8월 --> 8: 3

지출내용	지출액	날짜(2023MMDD)	카테고리
커피구매	8000	2023-03-05	식비
화장품구매	50000	2023-03-10	미용
전기요금	60000	2023-03-15	생활용품
운동화구매	90000	2023-03-20	쇼핑
의료진단비	75000	2023-03-25	의료/건강
식료품	48000	2023-03-30	식비

4. Changes in Comparison to the Plan

1) Output expenditure history

- Before: Enter year and month to output history
- After: Limit duration to months
- Reason: I tried to get the year and month intersection but failed because the datetime error continued.

5. Schedule

업무		11/3	11/26	12/8	12/22
제안서 작성		완료	완료	완료	
지출 저장			완료	수정 완료	
내역 출력	기간 지출액			완료	
	평균 지출액				
기능 모듈화					진행중