

5a3c8074-0943-4c32-8d67-0dbd0afc55c3

May 1, 2025

Olá, Nathália!

Meu nome é Luiz. Fico feliz em revisar seu projeto. Ao longo do texto farei algumas observações sobre melhorias no código e também farei comentários sobre suas percepções sobre o assunto. Estarei aberto a feedbacks e discussões sobre o tema.

Peço que mantenha e não altere os comentários que eu fizer por aqui para que possamos nos localizar posteriormente, ok?

Mais uma coisa, vamos utilizar um código de cores para você entender o meu feedback no seu notebook. Funciona assim:

Comentário do revisor:

Sucesso. Tudo foi feito corretamente.

Comentário do revisor:

Alerta não crítico, mas que pode ser corrigido para melhoria geral no seu código/análise.

Comentário do revisor:

Erro que precisa ser arrumado, caso contrário seu projeto **não** será aceito.

Você pode interagir comigo através dessa célula: Bom dia Luiz, tudo bem? Desculpa, mas como sou leiga no assunto, estou ten

Resposta do Aluno.

Comentário do revisor v1

Obrigado por enviar o seu projeto e pelo esforço para executar as etapas necessárias. Saber manipular corretamente as funções e operadores no python é muito relevante para uma carreira na área de dados.

No geral, o seu projeto está bom, você implementou corretamente a maioria das questões. Entretanto, deixei alguns comentários em questões que precisam de pequenos ajustes no código para avançarmos.

Se tiver alguma dúvida, pode contar comigo.

Até breve!

Comentário do revisor v2

Obrigado por enviar o seu projeto e pelo esforço para executar as etapas necessárias.

Essa versão do seu trabalho está muito melhor, parabéns. Entretanto, deixei alguns comentários em pontos que precisamos de pequenos ajustes para avançarmos. Peço que trabalhe nos comentários em vermelho que não foram riscados para seguirmos.

Se tiver alguma dúvida, pode contar comigo.

Até breve!

Comentário geral do revisor

Obrigado por enviar o seu projeto e fazer os ajustes sugeridos. Essa versão do seu trabalho ficou muito melhor! Espero que as sugestões sejam relevantes para projetos futuros.

Te desejo uma jornada de muito sucesso e aprendizado.

Qualquer dúvida, pode contar comigo.

Até breve!

Uma empresa de comércio eletrônico, Store 1, começou recentemente a coletar dados sobre seus clientes. O objetivo da Store 1 é entender melhor o comportamento dos clientes e tomar decisões baseadas em dados para melhorar experiência online deles.

Como parte da equipe analítica, sua primeira tarefa é avaliar a qualidade de uma amostra de dados coletados e preparar elas para análises futuras.

1 Quiz

A Store 1 visa garantir a consistência na coleta de dados. Como parte desse esforço, a qualidade dos dados coletados sobre os usuários precisa ser avaliada. Foi pedido que você revise os dados coletados e proponha alterações. Abaixo, você verá dados sobre um determinado usuário. Revise os dados e identifique possíveis problemas.

```
[ ]: user_id = '32415'
      user_name = ' mike_reed '
      user_age = 32.0
      fav_categories = ['ELECTRONICS', 'SPORT', 'BOOKS']
```

Opções:

1. O tipo de dados de `user_id` deve ser alterado de string para número inteiro (integer). Não, porém, existe contro versas. O ‘`user_id`’ é como fosse o RG ou CPF de uma pessoa. Caso for número interio, e conter zero, o zero é excluído à esquerda.
2. A variável `user_name` contém uma string com espaçamento desnecessário e um sublinhado entre o nome e o sobrenome. Sim, verdadeiro.
3. O tipo de dados de `user_age` está incorreto. Não, não está incorreto por estar em formato float, porém, para melhor indentificação a formatação para número inteiro melhora na descrição.
4. A lista `fav_categories` contém strings em letras maiúsculas. Em vez disso, devemos converter os valores da lista para letras minúsculas. Não há necessidade. Tanto letras maiúsculas ou minúsculas nesse caso estaria correto.

Escreva na célula Markdown abaixo o número de opções que você identificou como problemas. Se você identificou vários problemas, separe o número por vírgulas. Por exemplo, se você acha que os números 1 e 3 estão incorretos, escreva 1, 3, e explique o motivo.

Escreva sua resposta e explique seu raciocínio:

1. Não, porém, existe contro versas. O 'user_id' é como fosse o RG ou CPF de uma pessoa. Caso for número interio, e conter zero, o zero é excluído à esquerda.
2. Sim, verdadeiro.
3. O tipo de dados de user_age está incorreto. Não, não está incorreto por estar em formato float, porém, para melhor indentificação a formatação para número inteiro melhora na descrição.
4. Não há necessidade. Tanto letras maiúsculas ou minúsculas nesse caso estaria correto.

2 Tarefa 1

Vamos implementar as mudanças que identificamos. Primeiro, queremos corrigir os problemas com a variável `user_name`. Como verificamos, ela possui espaços desnecessários e um sublinhado como separador entre o nome e o sobrenome. Seu objetivo é remover os espaços e depois substituir o sublinhado por espaço.

```
[1]: user_name = ' mike_reed '  
user_name = user_name.strip()  
user_name = user_name.replace('_', ' ')  
  
print(user_name)
```

mike reed

Comentário do revisor:

Na questão acima, você precisa usar os métodos `strip()` e `replace()`, respectivamente.

3 Tarefa 2

Em seguida, precisamos dividir o `user_name` atualizado em duas substrings para obter uma lista que contém dois valores: a string para o nome e a string para o sobrenome.

```
[2]: user_name = 'mike reed'  
name_split = user_name.split()  
  
print(name_split)
```

['mike', 'reed']

Comentário do revisor:

Correto! O método `split()` permite quebrar uma string em um delimitador.

4 Tarefa 3

Ótimo! Agora queremos trabalhar com a variável `user_age`. Como mencionamos antes, ela possui um tipo de dados incorretos. Vamos corrigir esse problema transformando o tipo de dados e imprimindo o resultado final.

```
[3]: user_age = 32.0
      user_age = int(user_age)

      print(user_age)
```

32

Comentário do revisor:

Correto. O estudante converteu o tipo de dado do atributo “idade” para `int`.

5 Tarefa 4

Como sabemos, os dados nem sempre são perfeitos. Temos que considerar cenários em que o valor de `user_age` não pode ser convertido em um número inteiro. Para evitar que nosso sistema falhe, devemos tomar medidas com antecedência.

Escreva um código que tenta converter a variável `user_age` em um número inteiro e atribua o valor transformado a `user_age_int`. Se a tentativa falhar, vamos exibir uma mensagem solicitando que o usuário forneça sua idade como um valor numérico com a mensagem: **Forneça sua idade como um valor numérico**.

```
[25]: user_age = 'thirty two'

      try:
          user_age_int = int('user_age')

      except ValueError:
          print("Erro: Forneça sua idade como um valor numérico")
```

Erro: Forneça sua idade como um valor numérico

Comentário do revisor:

Aqui você poderia especificar qual tipo de exceção o seu código irá manipular. Adicione a exceção do tipo `ValueError` nesse caso.

Referência: https://www.w3schools.com/python/python_try_except.asp.

6 Tarefa 5

Por fim, observe que todas as categorias de favoritos são armazenadas em letras maiúsculas. Para preencher uma nova lista chamada `fav_categories_low` com as mesmas categorias, mas em letras minúsculas, repita os valores na lista `fav_categories`, os modifique e anexe os novos valores à lista `fav_categories_low`. Como sempre, imprima o resultado final.

```
[5]: fav_categories = ['ELECTRONICS', 'SPORT', 'BOOKS']
     fav_categories_low = []
     for cat in fav_categories:
         fav_categories_low.append(cat.lower())
     print(fav_categories_low)
```

```
['electronics', 'sport', 'books']
```

Comentário do revisor:

O trecho de código acima está com erro de execução. Você precisa percorrer a lista acima usando o ciclo `for`, por exemplo.

Sugestão usando list comprehensions:

```
[cat.lower() for cat in categories]
```

Referência: https://w3schools.com/python/python_lists_comprehension.asp

7 Tarefa 6

Conseguimos informações adicionais sobre os hábitos de consumo de nossos usuários, incluindo o valor gasto em cada uma de suas categorias favoritas. A administração está interessada nas seguintes métricas:

- Valor total gasto pelo usuário
- Valor mínimo gasto
- Valor máximo gasto

Vamos calcular e imprimir esses valores:

```
[6]: fav_categories_low = ['electronics', 'sport', 'books']
     spendings_per_category = [894, 213, 173]

     total_amount = sum(spendings_per_category)
     max_amount = min(spendings_per_category)
     min_amount = max(spendings_per_category)

     # não remova a instrução de impressão abaixo
     print(total_amount)
     print(max_amount)
     print(min_amount)
```

```
1280
```

```
173
```

894

Comentário do revisor:

Correto. O estudante calculou corretamente o valor total, mínimo e máximo.

8 Tarefa 7

A empresa quer oferecer descontos aos seus clientes fiéis. Clientes que fizerem compras totalizando mais de \$1.500 são considerados fiéis e vão receber um desconto.

Nosso objetivo é criar um ciclo `while` que verifique o valor total gasto e pare quando ele for atingido. Para simular novas compras, a variável `new_purchase` gera um número entre 30 e 80 em cada ciclo. Isso representa a quantidade de dinheiro gasto em uma nova compra, e é o que você precisa adicionar ao total.

Assim que o valor alvo for atingido e o ciclo `while` for encerrado, o valor final será impresso.

```
[12]: from random import randint

total_amount_spent = 1280
target_amount = 1500

while total_amount_spent < target_amount:

    new_purchase = randint(30, 80)
    total_amount_spent += new_purchase
print(total_amount_spent)
```

1523

Comentário do revisor:

Correto, porém o seu código possui problemas de indentação (perceba que tem uma “->” no bloco de código). Considere apagar essa seta e usar o `tab` novamente para indentação.

9 Tarefa 8

Agora temos todas as informações sobre um cliente da maneira que queremos. A administração de uma empresa nos pediu para encontrar uma maneira de resumir toda a informação sobre um usuário. Seu objetivo é criar uma string formatada que usa informações das variáveis `user_id`, `user_name` e `user_age`.

Aqui está a string final que queremos criar: Usuário 32415 chama-se mike e tem 32 anos.

```
[15]: user_id = '32415'
      user_name = ['mike', 'reed']
      user_age = 32

      print(f" Usuário {user_id} chama-se {user_name[0]} e tem {user_age} anos.")

      print(user_info)
```

Usuário 32415 chama-se mike e tem 32 anos.

Comentário do revisor:

Aqui você pode usar **f-strings** ou o método `.format()` para interpolar os valores com a string acima.

Como você já deve saber, as empresas coletam e armazenam dados de uma maneira específica. A Store 1 deseja armazenar todas as informações sobre seus clientes em uma tabela.

user_id	user_name	user_age	purchase_category	spending_per_category
'32415'	'mike', 'reed'	32	'electronics', 'sport', 'books'	894, 213, 173
'31980'	'kate', 'morgan'	24	'clothes', 'shoes'	439, 390

Em termos técnicos, uma tabela é simplesmente uma lista aninhada que possui uma sublista para cada usuário.

A Store 1 criou essa tabela para seus usuários. Ela está armazenada na variável `users`. Cada sublista contém o ID do usuário, nome e sobrenome, idade, categorias favoritas e o valor gasto em cada categoria.

10 Tarefa 9

Para calcular a receita da empresa, siga estas etapas:

1. Use um ciclo `for` para iterar na lista `users`.
2. Extraia a lista de gastos de cada usuário e some os valores.
3. Atualize o valor da receita com o total de cada usuário.

Isso vai fornecer a receita total da empresa, que você vai imprimir no final.

```
[19]: users = [
      # este é o começo da primeira sublista
      ['32415', ['mike', 'reed'], 32, ['electronics', 'sport', 'books'],
      [894, 213, 173]
      ], # este é o fim da primeira sublista

      # este é o começo da segunda sublista
```

```

    ['31980', ['kate', 'morgan'], 24, ['clothes', 'shoes'],
      [439, 390]
  ] # este é o fim da segunda sublista
]

revenue = 0

for user in users:
    spendings_list = user[4]
    total_spendings = sum(spendings_list)
    revenue += total_spendings

print(revenue)

```

2109

Comentário do revisor:

Aqui você precisa somar os itens do último elemento de cada lista acima e, posteriormente, realizar a soma total. O resultado esperado é 2109.

Comentário do revisor v2:

Sugestão:

```

for user in users:
    spendings_list = user[4]
    total_spendings = sum(...) # complete o código
    revenue += total_spendings

```

11 Tarefa 10

Use um ciclo for para percorrer a lista de usuários que fornecemos e imprima os nomes dos clientes com menos de 30 anos.

```

[22]: users = [
    ['32415', ['mike', 'reed'], 32, ['electronics', 'sport', 'books'],
      [894, 213, 173]],
    ['31980', ['kate', 'morgan'], 24, ['clothes', 'books'], [439,
      390]],
    ['32156', ['john', 'doe'], 37, ['electronics', 'home', 'food'],
      [459, 120, 99]],
    ['32761', ['samantha', 'smith'], 29, ['clothes', 'electronics',
      'beauty'], [299, 679, 85]],
    ['32984', ['david', 'white'], 41, ['books', 'home', 'sport'], [234,
      329, 243]],
    ['33001', ['emily', 'brown'], 26, ['beauty', 'home', 'food'], [213,
      659, 79]],

```



```

['33767', ['maria', 'garcia'], 33, ['clothes', 'food', 'beauty'],
 [499, 189, 63]],
['33912', ['jose', 'martinez'], 22, ['sport', 'electronics', 'home'
 ], [259, 549, 109]],
['34009', ['lisa', 'wilson'], 35, ['home', 'books', 'clothes'],
 [329, 189, 329]],
['34278', ['james', 'lee'], 28, ['beauty', 'clothes', 'electronics'
 ], [189, 299, 579]],
]

for user in users:
    if user[2] < 30:
        print(user[1][0])

```

kate
 samantha
 emily
 jose
 james

Comentário do revisor:

Aqui você precisa imprimir apenas o nome dos usuários. Dica: veja que o nome do usuário está na posição 1 da lista acima.

12 Tarefa 11

Vamos juntar as tarefas 9 e 10 e imprimir os nomes de usuários com menos de 30 anos com gastos totais acima de 1.000 dólares.

```

[23]: users = [
    ['32415', ['mike', 'reed'], 32, ['electronics', 'sport', 'books'],
     [894, 213, 173]],
    ['31980', ['kate', 'morgan'], 24, ['clothes', 'books'], [439,
     390]],
    ['32156', ['john', 'doe'], 37, ['electronics', 'home', 'food'],
     [459, 120, 99]],
    ['32761', ['samantha', 'smith'], 29, ['clothes', 'electronics',
     'beauty'], [299, 679, 85]],
    ['32984', ['david', 'white'], 41, ['books', 'home', 'sport'], [234,
     329, 243]],
    ['33001', ['emily', 'brown'], 26, ['beauty', 'home', 'food'], [213,
     659, 79]],
    ['33767', ['maria', 'garcia'], 33, ['clothes', 'food', 'beauty'],
     [499, 189, 63]],
    ['33912', ['jose', 'martinez'], 22, ['sport', 'electronics', 'home'
     ], [259, 549, 109]],
    ['34009', ['lisa', 'wilson'], 35, ['home', 'books', 'clothes'],

```

```

    [329, 189, 329]],
    ['34278', ['james', 'lee'], 28, ['beauty', 'clothes', 'electronics'
], [189, 299, 579]],
]

```

```

for user in users:
    spendings_list = user[4]
    total_spendings = sum(spendings_list)

    if user[2] < 30 and total_spendings > 1000:
        print(user[1] [0])

```

samantha

james

Comentário do revisor:

Mesma observação da questão anterior.

13 Tarefa 12

Agora vamos imprimir o nome e a idade de todos os usuários que compraram roupas (clothes). Imprima o nome e a idade na mesma instrução de impressão.

```

[24]: users = [
    ['32415', ['mike', 'reed'], 32, ['electronics', 'sport', 'books'],
    [894, 213, 173]],
    ['31980', ['kate', 'morgan'], 24, ['clothes', 'books'], [439,
    390]],
    ['32156', ['john', 'doe'], 37, ['electronics', 'home', 'food'],
    [459, 120, 99]],
    ['32761', ['samantha', 'smith'], 29, ['clothes', 'electronics',
    'beauty'], [299, 679, 85]],
    ['32984', ['david', 'white'], 41, ['books', 'home', 'sport'], [234,
    329, 243]],
    ['33001', ['emily', 'brown'], 26, ['beauty', 'home', 'food'], [213,
    659, 79]],
    ['33767', ['maria', 'garcia'], 33, ['clothes', 'food', 'beauty'],
    [499, 189, 63]],
    ['33912', ['jose', 'martinez'], 22, ['sport', 'electronics', 'home'
], [259, 549, 109]],
    ['34009', ['lisa', 'wilson'], 35, ['home', 'books', 'clothes'],
    [329, 189, 329]],
    ['34278', ['james', 'lee'], 28, ['beauty', 'clothes', 'electronics'
], [189, 299, 579]],
]

```

```
for user in users:
    if 'clothes' in user[3]:
        print(user[1][0], user[2])
```

#escreva seu código aqui

kate 24
samantha 29
maria 33
lisa 35
james 28

Comentário do revisor:

Aqui você pode usar o operador `in` para verificar se a categoria `clothes` contém na lista de categorias em que o usuário fez compras. Sugestão:

```
# adapte para o seu código
if `clothes` in categorias:
    ...
```

#Escreva quaisquer comentários ou ideias finais aqui