

PROYECTO 2 PYTHON BASICO

PATRICIA PASTRANA

DESCRIPCIÓN DEL PROYECTO

Todavía desempeñándote como miembro del equipo analítico, en el primer proyecto hemos sentado las bases para la segunda fase. ¡Hemos llegado! Ahora aplicarás técnicas avanzadas para extraer datos significativos, atendiendo a las crecientes necesidades del cliente.

Como sabes, las empresas recopilan y almacenan datos de una forma particular. Store 1 quiere almacenar toda la información de sus clientes en una tabla.

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

En términos técnicos, una tabla es simplemente una lista anidada que contiene una sublista para cada usuario o usuaria.

Store 1 ha creado una tabla de este tipo para sus usuarios. Está almacenada en la variable "users". Cada sublista contiene el ID del usuario, nombre y apellido, edad, categorías favoritas y el importe gastado en cada categoría.

- **user_id:** el identificador único para cada usuario.
- **user_name:** el nombre de usuario.
- **user_age:** la edad del usuario.
- **fav_categories:** las categorías de artículos comprados por el usuario, como 'ELECTRONICS', 'SPORT', 'BOOKS', etc.
- **total_spendings:** la lista de enteros que indican la cantidad gastada en cada una de sus categorías favoritas.

```
In [22]: users = [  
    ['32415', ' mike_reed ', 32.0, ['ELECTRONICS', 'SPORT', 'BOOKS'], [894, 213, 173]],  
    ['31980', 'kate morgan', 24.0, ['CLOTHES', 'BOOKS'], [439, 390]],  
    ['32156', ' john doe ', 37.0, ['ELECTRONICS', 'HOME', 'FOOD'], [459, 120, 99]],  
    ['32761', 'SAMANTHA SMITH', 29.0, ['CLOTHES', 'ELECTRONICS', 'BEAUTY'], [299, 679, 85]],  
    ['32984', 'David White', 41.0, ['BOOKS', 'HOME', 'SPORT'], [234, 329, 243]],  
    ['33001', 'emily brown', 26.0, ['BEAUTY', 'HOME', 'FOOD'], [213, 659, 79]],  
    ['33767', ' Maria Garcia', 33.0, ['CLOTHES', 'FOOD', 'BEAUTY'], [499, 189, 63]],  
    ['33912', 'JOSE MARTINEZ', 22.0, ['SPORT', 'ELECTRONICS', 'HOME'], [259, 549, 109]],  
    ['34009', 'lisa wilson ', 35.0, ['HOME', 'BOOKS', 'CLOTHES'], [329, 189, 329]],  
    ['34278', 'James Lee', 28.0, ['BEAUTY', 'CLOTHES', 'ELECTRONICS'], [189, 299, 579]],  
]
```

Paso 1

En la última tarea de la primera parte de este proyecto escribiste código para:

1. Eliminar todos los espacios iniciales y finales de los nombres, así como cualquier guion bajo.
2. Convertir todas las edades en números enteros.
3. Separar todos los nombres y apellidos en una sublista.

Hagámoslo ahora una función para que podamos usarla para fijar a cualquier cliente. Nombra a tu función `clean_user`. Debe recibir una lista con toda la información del cliente (`user_info`), así como dos enteros. Uno de ellos señala el índice del nombre del cliente y el otro es el índice de la edad del cliente en la lista. Debe devolver la lista limpia después de aplicar todos los cambios anteriores. Pruébala llamándola, pasándole la lista `test_user[]` y luego muéstrala en pantalla.

```
In [23]: # define tu función aquí
def clean_user (call_user, name, age):

    # Paso 1: elimina del nombre espacios iniciales y finales, así como guiones

    temporal = call_user[name].strip().replace('_', ' ')

    # Paso 2: convierte la edad en entero
    user_age_1 = int(call_user[age])

    # Paso 3: separa el nombre y el apellido en una sublista
    user_name_1 = temporal.split()

    # Prepara la lista con la información completa del usuario
    # Reemplaza el nombre y la edad originales con los datos limpios
    call_user[name] = user_name_1
    call_user[age] = user_age_1

    return call_user

# Prueba la función
test_user = ['32415', ' mike_reed ', 32.0, ['ELECTRONICS', 'SPORT', 'BOOKS'], [894, 213, 173]]
name_index = 1
age_index = 2

print(clean_user(test_user, name_index, age_index))
```

```
['32415', ['mike', 'reed'], 32, ['ELECTRONICS', 'SPORT', 'BOOKS'], [894, 213, 173]]
```

Paso 2

Observa que todas las categorías favoritas están almacenadas en mayúsculas. Llena una nueva lista llamada `fav_categories_low` con las mismas categorías, pero en minúsculas, iterando sobre los valores en la lista `fav_categories`, modificándolos y luego añade los nuevos valores a la lista `fav_categories_low`. Como siempre, muestra el resultado final.

```
In [24]: fav_categories = ['ELECTRONICS', 'SPORT', 'BOOKS']
fav_categories_low = []

# escribe tu código aquí
```

```

for fav_categorie in fav_categories:
    fav_categories_low.append(fav_categorie.lower())

print(fav_categories_low)
#como puedo hacerle par que mi lista se vea de la siguiente manera ['Electronics','Sport','Books
['electronics', 'sport', 'books']

```

Paso 3

Ahora hagamos lo mismo, pero para cada uno de los usuarios de la empresa. Llena una lista nueva llamada `users_categories_low` con los mismos usuarios, pero con sus categorías en minúsculas, iterando sobre los valores en la lista `users`, luego itera sobre los valores en `user_categories`, modificándolos, y después agrega los nuevos valores de usuarios a la lista `users_categories_low`. Como siempre, muestra el resultado final.

```

In [25]: users = [
    ['32415', ' mike_reed ', 32.0, ['ELECTRONICS', 'SPORT', 'BOOKS'], [894, 213, 173]],
    ['31980', 'kate morgan', 24.0, ['CLOTHES', 'BOOKS'], [439, 390]],
    ['32156', ' john doe ', 37.0, ['ELECTRONICS', 'HOME', 'FOOD'], [459, 120, 99]],
    ['32761', 'SAMANTHA SMITH', 29.0, ['CLOTHES', 'ELECTRONICS', 'BEAUTY'], [299, 679, 85]],
    ['32984', 'David White', 41.0, ['BOOKS', 'HOME', 'SPORT'], [234, 329, 243]],
    ['33001', 'emily brown', 26.0, ['BEAUTY', 'HOME', 'FOOD'], [213, 659, 79]],
    ['33767', ' Maria Garcia', 33.0, ['CLOTHES', 'FOOD', 'BEAUTY'], [499, 189, 63]],
    ['33912', 'JOSE MARTINEZ', 22.0, ['SPORT', 'ELECTRONICS', 'HOME'], [259, 549, 109]],
    ['34009', 'lisa wilson ', 35.0, ['HOME', 'BOOKS', 'CLOTHES'], [329, 189, 329]],
    ['34278', 'James Lee', 28.0, ['BEAUTY', 'CLOTHES', 'ELECTRONICS'], [189, 299, 579]],
]

users_categories_low = []
for user in users:
    categories_low = []
    for category in user[3]:
        lowered_category = category.lower()
        categories_low.append(lowered_category)
    user.pop(3)
    user.insert(3, categories_low)
    users_categories_low.append(user)
print(users_categories_low)

```

```
[[ '32415', ' mike_reed ', 32.0, ['electronics', 'sport', 'books'], [894, 213, 173]], [ '31980', 'k
ate morgan', 24.0, ['clothes', 'books'], [439, 390]], [ '32156', ' john doe ', 37.0, ['electronic
s', 'home', 'food'], [459, 120, 99]], [ '32761', 'SAMANTHA SMITH', 29.0, ['clothes', 'electronic
s', 'beauty'], [299, 679, 85]], [ '32984', 'David White', 41.0, ['books', 'home', 'sport'], [234,
329, 243]], [ '33001', 'emily brown', 26.0, ['beauty', 'home', 'food'], [213, 659, 79]], [ '33767',
' Maria Garcia', 33.0, ['clothes', 'food', 'beauty'], [499, 189, 63]], [ '33912', 'JOSE MARTINEZ',
22.0, ['sport', 'electronics', 'home'], [259, 549, 109]], [ '34009', 'lisa wilson ', 35.0, ['hom
e', 'books', 'clothes'], [329, 189, 329]], [ '34278', 'James Lee', 28.0, ['beauty', 'clothes', 'el
ectronics'], [189, 299, 579]]]
```

Paso 4

Ahora, completemos el código de nuestra función `clean_user` para limpiar la categoría:

1. Añade otro parámetro con el índice de categorías.
2. Pon todos los nombres en minúsculas antes de aplicar "strip" y "replace".

Después, crea un bucle y aplica tu función a toda la lista de usuarios, agregando tus resultados a la lista `users_clean`. Después muéstralo

```
In [26]: def clean_user(user_info, name_index, age_index, cat_index):

    # Paso 1: pon todo en minúsculas y elimina del nombre espacios iniciales y finales, así como
    user_name_1 = user_info[name_index].lower().strip().replace('_', ' ')

    # Paso 2: convierte la edad en entero
    user_age_1 = int(user_info[age_index])
    # Paso 3: separa el nombre y el apellido en una sublista
    user_name_1 = user_name_1.split()
    # Paso 4: pon categoría a minúsculas
    users_clean = []

    for category in user_info[cat_index]:
        lowered_category = category.lower()
        users_clean.append(lowered_category)

    # Prepara la lista con la información completa del usuario
    # Reemplaza el nombre y la edad originales con los datos limpios
    # escribe tu código aquí

    user_info[name_index] = user_name_1
    user_info[age_index] = user_age_1
    user_info[cat_index] = users_clean

    return user_info

users = [
    [ '32415', ' mike_reed ', 32.0, ['ELECTRONICS', 'SPORT', 'BOOKS'], [894, 213, 173]],
    [ '31980', 'kate morgan', 24.0, ['CLOTHES', 'BOOKS'], [439, 390]],
    [ '32156', ' john doe ', 37.0, ['ELECTRONICS', 'HOME', 'FOOD'], [459, 120, 99]],
    [ '32761', 'SAMANTHA SMITH', 29.0, ['CLOTHES', 'ELECTRONICS', 'BEAUTY'], [299, 679, 85]],
    [ '32984', 'David White', 41.0, ['BOOKS', 'HOME', 'SPORT'], [234, 329, 243]],
    [ '33001', 'emily brown', 26.0, ['BEAUTY', 'HOME', 'FOOD'], [213, 659, 79]],
    [ '33767', ' Maria Garcia', 33.0, ['CLOTHES', 'FOOD', 'BEAUTY'], [499, 189, 63]],
    [ '33912', 'JOSE MARTINEZ', 22.0, ['SPORT', 'ELECTRONICS', 'HOME'], [259, 549, 109]],
```

```

['34009', 'lisa wilson ', 35.0, ['HOME', 'BOOKS', 'CLOTHES'], [329, 189, 329]],
['34278', 'James Lee', 28.0, ['BEAUTY', 'CLOTHES', 'ELECTRONICS'], [189, 299, 579]],
]

name_index = 1
age_index = 2
cat_index = 3
users_cleaned = []

for user in users:    # escribe tu código aquí

    user_cleaned = clean_user(user, name_index, age_index, cat_index)
    users_cleaned.append(user_cleaned)

for x in users_cleaned:

    print(x)

```

```

['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]]

```

Paso 5

La empresa desea conocer sus ingresos totales y te pide que proporciones este valor. Para calcular los ingresos de la empresa, sigue estos pasos:

1. Utiliza `for` para iterar sobre la lista `users`.
2. Extrae la lista de gastos de cada usuario y suma los valores.
3. Actualiza el valor de los ingresos con el total de cada usuario.

Así obtendrás los ingresos totales de la empresa que mostrarás en la pantalla al final.

```

In [27]: 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]]]

revenue = 0

```

```

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

    #revenue += sum(user[4])

print(revenue)

```

9189

Hola, de acuerdo a los comentarios trate de corregir lo que me decia y crear las variables, pero ahora mi resultado es ese, la verdad no encuentro donde esta mal y porque NO me hace la suma, no entiendo.. :(

Paso 6

La empresa quiere ofrecer descuentos a sus clientes leales. Los clientes que realizan compras por un importe total mayor a \$1500 se consideran leales y recibirán un descuento.

Nuestro objetivo es crear un bucle `while` que compruebe el importe total gastado y se detenga al alcanzarlo. Para simular nuevas compras, la variable `new_purchase` genera un número entre 30 y 80 en cada iteración del bucle. Esto representa la cantidad de dinero gastada en una nueva compra y es lo que debes agregar al total.

Una vez que se alcance el importe objetivo y se termine el bucle `while`, se mostrará la cantidad final.

```

In [ ]: from random import randint

total_amount_spent = 1280
target_amount = 1500

while total_amount_spent < target_amount: # escribe tu código aquí
    new_purchase = randint(30, 80) # generamos un número aleatorio de 30 a 80
    total_amount_spent += new_purchase # escribe tu código aquí

print(total_amount_spent)

```

1529

Paso 7

Recorre la lista de usuarios que te hemos proporcionado y muestra los nombres de los clientes menores de 30 años.

```

In [ ]: 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, 800]],
                ['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])
```

```
['kate', 'morgan']
['samantha', 'smith']
['emily', 'brown']
['jose', 'martinez']
['james', 'lee']
```

Paso 8

Mostremos en pantalla los nombres de los usuarios menores de 30 años que acumulan un gasto total superior a 1000 dólares.

```
In [ ]: 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, 894]],
                ['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]]]

# escribe tu código aquí
for user in users:
    if user[2] < 30:
        result = sum(user[4])
    if user[2] < 30 and result > 1000:
        print(user[1])
```

```
['samantha', 'smith']
['james', 'lee']
```

Paso 9

Ahora vamos a mostrar el nombre y la edad de todos los usuarios y todas las usuarias que han comprado ropa ("clothes"). Imprime el nombre y la edad en la misma declaración print.

```
In [ ]: 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, 894]],
                ['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]]

# escribe tu código aquí

clos = 'clothes'

for user in users:
    for clos in user[3]:
        if clos in 'clothes':

            #if clothes in users[3]:
                print(user[1], user[2])

            #users = user[1][2]

#print(user)

```

```

['kate', 'morgan'] 24
['samantha', 'smith'] 29
['maria', 'garcia'] 33
['lisa', 'wilson'] 35
['james', 'lee'] 28

```

*******Pista*******

Utiliza el bucle for para iterar sobre cada fila de la tabla. A continuación, utiliza otro bucle para comprobar si este usuario ha comprado ropa ("clothes") y, en caso afirmativo, muestra el nombre y la edad dentro de la misma declaración print, separándolos por comas: `print(firstname, age)`.

Paso 10

La dirección requiere de una función que proporcione información sobre los clientes, incluyendo sus nombres, edades y gasto total, filtrada por categorías específicas. Con base en fragmentos de código anteriores, crearemos una función llamada `get_client_by_cat` con las siguientes especificaciones:

1. Parámetros:

- **users:** una lista con los datos de los usuarios.
- **id_index:** el índice donde está almacenado el ID del cliente en la lista de usuarios.
- **name_index:** el índice donde está almacenado el nombre del cliente en la lista de usuarios.
- **age_index:** el índice donde la edad del cliente está almacenada en la lista de usuarios.
- **category_index:** el índice donde las categorías de compras del cliente están listadas.

- **amounts_index:** el índice donde las cantidades gastadas en cada categoría están almacenadas.
- **filter_category:** un string que representa el nombre de la categoría para filtrar clientes.

2. Salida:

- La función devuelve una lista de sublistas. Cada sublista contiene:
 - El número ID del cliente.
 - Una sublista con el nombre y apellido del cliente.
 - La edad del cliente.
 - Un entero que representa la cantidad total gastada por el cliente.

Por ejemplo, si llamas a la función con los siguientes parámetros:

```
get_client_by_cat([
    ['32415', ['mike', 'reed'], 32, ['electronics', 'sport', 'books'], [894, 213,
173]]
], 0, 1, 2, 3, 4, 'books')
```

La salida será:

```
[['32415', ['mike', 'reed'], 32, 1280]]
```

Esta salida muestra que el cliente con el ID '32415', llamado Mike Reed, de 32 años, gastó un total de 1280 en la categoría 'books' y otras compras.

Después de hacer tu función, llámala pasándole nuestra lista de usuarios, los índices adecuados y la categoría 'home' y muestra en pantalla la lista que resulta.

```
In [ ]: def get_client_by_cat(id_index, name_index, age_index, category_index, amounts_index, filter_cat
sub_client = []
clients_filter = []

for user in users:
    total_gastos = sum(user[amounts_index])
    if filter_category in user[category_index]:
        sub_client = [user[id_index], user[name_index], user[age_index], user[category_index]]
        clients_filter.append(sub_client)
return(clients_filter)

# escribe tu código aquí

# La lista de usuarios
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]],git
    ['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]]
]

# Llama a la función con la categoría 'home'
#result = get_client_by_cat([id_index, name_index, age_index, category_index, amounts_index],0,1,
result = get_client_by_cat(0, 1, 2, 3, 4, 'home')
```

```
# Muestra en pantalla la lista que resulta  
print(result)
```

```
[['32156', ['john', 'doe'], 37, ['electronics', 'home', 'food'], 678], ['32984', ['david', 'white'], 41, ['books', 'home', 'sport'], 806], ['33001', ['emily', 'brown'], 26, ['beauty', 'home', 'food'], 951], ['33912', ['jose', 'martinez'], 22, ['sport', 'electronics', 'home'], 917], ['34009', ['lisa', 'wilson'], 35, ['home', 'books', 'clothes'], 847]]
```