Develop a secured back-end architecture with Python and SQL

0A. Background

Epic Events, a company that organizes "epic parties" for start-ups, asked for a Customer Relationship Management (CRM) program to regain their customers' trust, as their current tools aren't adapted to their needs.

That secured system would allow them to collect and process customers' as well as their respective events' data.

The main goal is to set a database that would allow them to store and process data about the customers, the contracts and the events in a secured way. The program must be made in Python and must have Sentry implemented in order to run a log and monitor errors and some specific events.

0B. Installation

Beware: the installation of PostgreSQL is mandatory. Please do so **before** trying to run the program.

A recommended website with step-by-step instructions (also available for macOS and Linux):
https://www.postgresqltutorial.com/postgresql-getting-started/install-postgresql/
The official website: https://www.postgresql.org/
Go to the Download section and choose the version that suits your OS.

Important: if ou chose to use pgAdmin 4 (like I did) and your program seems to be stuck in a loop right after starting it, download and install a previous version. There seems to be serious issues with the latest versions (as of July 2023), I had to install v7.1 to get it to run.

You can also choose another client rather than pgAdmin.

Once you've done that, back to my program:

Clone the repository (or download it directly):

```
git clone https://github.com/munchou/OpenClassrooms-Project-12.git
```

Go to the program's directory and create a virtual environment:

`cd OpenClassrooms-Project-12` (or any folder where the project is located)

`python -m venv _env` where _env is the name of the folder where the environment will be created.

Activation: `_env/Scripts/activate`

Install the needed modules:

```
pip install -r requirements.txt
```

1. Start the program

Absolutely **everything** goes through the program. Yes, EVE-RY-THING! Which means unlike most things you'll find, you do not need to run ANY commands to create your database or connect to it. Some would say it's strange or not even recommended, and I would disagree. The thing is: ONLY the superuser (the user that was created when PostgreSQL was installed) has the rights to do so. Any other user is NOT supposed to know the username, let alone the password to connect to the databases. So be it through psql or within my program, as long as you got the right IDs, you can do whatever you want.

Still in the command window where you activated your virtual environment, type:

```
python -m main
```

2. First launch: no database existing, no INI file

This is a cool part I am quite proud of. The program detects if the INI file exists (which is the config file from which the pogram gets the information to connect to the right database) and will prompt you to create it if it is not found.

So, the program should have started and you should be having something like that:

```
Configuration file not found. Let's create one.
IMPORTANT: Leave empty for default value (simply press ENTER)
Unless specified, it is strongly advised to use the default settings. You only must enter your password.

        Host (default: localhost):
```

You are the ADMIN, so you obviously know how to fill in those fields.

It is necessary to type in your password twice, any mistake would give birth to a defective config file, and we don't want that, do we?

```
        Host (default: localhost):
        Port (default: 5432):
        Database's name (default: postgres):
        User (default: postgres):
        PASSWORD (the one used when PostgreSQL was installed):
        Repeat the password:
The passwords did not match, please try again.
        PASSWORD (the one used when PostgreSQL was installed):
        Repeat the password:

Please confirm the configuration details:
Host: localhost
Port: 5432
Database's name: postgres
User: postgres
Password: ******

Confirm? (y/n) _
```

In the unlikely case that you did make a mistake, I fed my program with extra neurones to prevent any undesired destruction of the Universe. Because it cannot connect to the database, that smarty will display an error message AND remove the file that was just created. So what to do, you ask, panicking behind your monitor? Nothing, just press ENTER to confirm, which will close the program. You will need to restart it, and type in the correct information.

```
Confirm? (y/n) y
There are one or several errors in the provided connection's parameters.
Deleting the configuration file. Please restart the program and create a new config file.
Press ENTER to quit: _
```

Now, say you entered the correct information. You could connect to the default *postgres* database. However, I chose to forbid the user from modifying it in any way, so you will be asked to create your own database. To avoid compatibility issues would you choose one day to migrate your database to another system, it is advised to use ascii characters, ideally simply lowercase alphabetical characters. I also limited the length to 20 characters, which should not be a problem.
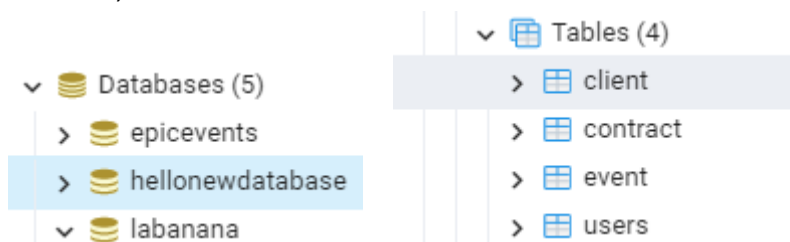
```
Because the current database is 'postgres', you must choose another one or create a new database.
*****************
Current database: postgres
*****************
- Please enter the name of the database you wish to load or create.
- Leave blank and simply press ENTER to use the current database ('postgres').
- Regarding CREATION:
        Only alphabet characters (letters aA, bB, cC...), digits (0, 1, 2...), and [@ _ $ #].
        NO SPACES ALLOWED!
        Maximum length: 20 characters

        database: _
```

Two things can happen here: either you type in a database that does not exist on your machine or one that already exists.

Case one: you type "hellonewdatabase", the program detects that it does not exist. So not only will it create it, it will also create the tables.

```
∨ ⊞ Tables (4)
    > ⊞ client
    > ⊞ contract
    > ⊞ event
    > ⊞ users

∨ ⬛ Databases (5)
    > ⬛ epicevents
    > ⬛ hellonewdatabase
    ∨ ⬛ labanana
```

Case two: the database already existed. For example, in my case, I have been working with one named "labanana". Instead of trying to rewrite or erase anything, the program will detect the existing database and connect to it. It **obviously** has to be a database that matches that CRM program...

So this is what I would get:

```
* * * * * * * * * * * * * * * * * * * * *
* AUTHENTICATION to database 'labanana' *
* * * * * * * * * * * * * * * * * * * * *
(type 'exit' as the username to exit the program)

        username:
```

3. Admin, create the first user

Remember, you are the ADMIN, the SUPERUSER, and even though you don't have much to do with the employees of the company who will then be using that program and finally understand what happiness is, you've got to set it up so that they can start doing something with it. All you have to do is create a staff member belonging to the Management team.

To do so, simply connect by typing in your IDs.

```
* * * * * * * * * * * * * * * * * * * * * * * * *
* AUTHENTICATION to database 'hellonewdatabase' *
* * * * * * * * * * * * * * * * * * * * * * * * *
(type 'exit' as the username to exit the program)

        username: postgres
        password:
```

You will then be transported to a new world full of possibilities. Yes, actually, as the admin, you have full powers (except for one thing), so don't start messing around, it's only in case of emergency!

The only thing we are interested in is *1. Create a user*.

```
Hello, God. What will it be today?
        db_update. Create/Update a database
        db_removetables. Remove all the database's table

        1. Create a user (management team)
        2. Update a user (management team)
        3. Deactivate a user (management team)
        4. Create a client (sales team)
        5. Update a client (salesman in charge of the client)
        6. Display all clients (all the teams)
        7. Create a contract (management team)
        8. (unavailable for admin) Update a contract (management + salesman in charge of the client)
        9. Display all the contracts (all the teams)
        10. Display contracts that haven't been signed yet (sales team)
        11. Display contracts that haven't been fully paid (sales team)
        12. Create an event (salesman in charge of the client)
        13. Display events without support staff (management team)
        14. Update an event to add support staff (management team)
        15. Update an event (support member in charge of it)
        16. Display events the support member is in charge of (support)
        17. Display all the events (all the teams)

        disconnect. DISCONNECT and go back to authentication
Your choice: 1
```

```
CREATION OF A USER:
        username: Manager1
        Password (at least 8 characters):
        Repeat the password:
        Full name: Manager One
        email (optional, press ENTER if no email): m1@hnd.com

        Telephone number: 911

Which team will the user be part of?
        1. Management Team
        2. Sales Team
        3. Support Team
CHOICE: 1
```

Apart from the password, the rest can be modified by the team, so choose wisely (or ask the employee what he wishes).

Before creating the new user, you will be asked to review the information and then confirm.

```
CREATION OF A USER:
        username: Manager1
        Password (at least 8 characters):
        Repeat the password:
        Full name: Manager One
        email (optional, press ENTER if no email): m1@hnd.com

        Telephone number: 911

Which team will the user be part of?
        1. Management Team
        2. Sales Team
        3. Support Team
CHOICE: 1

Please confirm the new user's details:
        Username: Manager1
        Full name: Manager One
        email: m1@hnd.com
        Phone number: 911
        Team: management

Confirm? (y/n)
```

```
Confirm? (y/n)
y
User Manager1 was successfully created.
        Press ENTER to go back to your menu.
```

Upon confirmation, you will be redirected to the main menu after pressing ENTER.

You can verify that the user is now inside the database by checking with SELECT * FROM users; in your client. With pgAdmin, you should get something like that:

	id [PK] integer	username character varying (50)	password character varying (100)	full_name character varying (100)
1	1	Manager1	{70e39f1487b006564cfe4…	Manager One

email character varying (100)	phone_number character varying (20)	status statusenum	saltychain character varying (200)
m1@hnd.com	911	management	{7873e171dbdd,d70671e522,40bcc45458}

Your job is done, well done! Now the manager can create, modify and deactivate users.

4. General use and common functionalities (for all teams)

Connection: simply type in your username and password. If you made a mistake, an error message will be displayed and you will be asked to type in again.

If your account has been deactivated, you will not be able to log in.

```
* * * * * * * * * * * * * * * * * * * * * *
* AUTHENTICATION to database 'labanana' *
* * * * * * * * * * * * * * * * * * * * * *
(type 'exit' as the username to exit the program)

        username: _
```

Menus: the menus differ depending on which team you belong to.

Important: before EVERY action that can modify the database in any way, you will be asked to type your password again. Yes, it's annoying, but it's a safety measure, would you forget to disconnect.

Disconnection: either directly close the program or, ideally type in "disconnect" on your menu page to be disconnected and redirected to the authentication page.

```
        disconnect. DISCONNECT and go back to authentication
Your choice: disconnect_
```

Closing the program: you can close it by closing the working window, or type in "exit" on the authentication page.

```
* * * * * * * * * * * * * * * * * * * * * *
* AUTHENTICATION to database 'labanana' *
* * * * * * * * * * * * * * * * * * * * * *
(type 'exit' as the username to exit the program)

        username: exit
```

Common functionalities: all teams can display the clients, contracts and events stored in the database. To do so, choose the corresponding option on your menu.

```
        8. Display all the clients
        9. Display all the contracts
        10. Display all the events

        disconnect. DISCONNECT and go back to authentication
Your choice: 9_
```

5. Management Team

```
MANAGEMENT MENU | WELCOME, Manager1
* * * * * * * * * * * * * * *
        1. Create a user
        2. Update a user
        3. Deactivate a user
        4. Create a contract
        5. Update a contract
        6. Display events without assigned support
        7. Assign support to an event
        8. Display all the clients
        9. Display all the contracts
        10. Display all the events

        disconnect. DISCONNECT and go back to authentication
Your choice: _
```

Hello Manager1. As you may have guessed, you are the first user of the program, the Patient 0, the solid base of that wonderful 800-meter tall skycraper, the future of your company. No pressure!

I assume you know how the organization of the teams of the company works (if you don't, I'd recommend you to start looking for a new job), so I won't explain about what each team is supposed to have access to or not.

● **To create a user**, see the previous section (section 4).

● **To update a user**, you must type in the ID of the user you wish to modify and confirm that you do want to modify their data. So let's start by changing your name, or else no one will know who you are.

```
Please enter the ID of the user you wish to update: 21
That user does not exist.
Please enter the ID of the user you wish to update: 1

You are about to update 'Manager1'.
Please confirm (y/n):
```

```
What would you like to update?
        1: username
        2: User's name
        3: email
        4: Phone number
        5: Status
        password: password
Choice: 2
```

```
Choice: 2
        Full name: Manny Kur
The user was updated successfully.
        Press ENTER to go back to your menu.
```

There, that's done, you now got your real name engraved inside the company's database! Although seeing your name, I suspect you may be trolling...

g (100)	full_name character varying (100)	email character varying
06564cfe4b...	Manny Kur	m1@hnd.com

Note: updating your own status will close your session and send you back to the authentication menu.

● **To deactivate a user**: it is NOT possible to delete a user (to prevent fraud, loss of important data, etc.), but a user can be deactivated by simply changing their status to one that does not allow access to the program's functionalities.

```
Please enter the ID of the user you wish to deactivate: 2
Do you confirm the deactivation of Manager2 (ID: 2) (y/n)? y_
```

Once it is done, the user that was deactivated, in our case *Manager2*, will not be able to connect anymore.

● **To reactivate a user**, a manager (or the admin) must simply update the target user's status.

● **To create a contract**, you must ensure that a client exists. If not, only a member of the Sales Team can create a client. Choose the *Create a contract* option and fill in the fields.
The contract being linked to its client, you cannot choose the salesman in charge of it. To do so, you must update the client's salesman in charge of them. The associated contract(s) of the client will then be automatically updated to match the new salesman in charge.

```
CREATION OF A CONTRACT:
        Client's ID: 3
        Contract's total price: 32000
        Contract's due amount: 12000
        The contract has been signed (y/n): y

Please confirm the new contract's details:
        Client's ID: 3
        Salesman in charge of the contract: 7
        Contract's total amount: 32000
        Due amount: 12000
        The contract has been signed.

Confirm? (y/n)
```

Note about the contract's signature: once a contract has been confirmed as signed, you **cannot** edit it to mark it as not signed (for obvious reasons). If you mark it as signed by mistake (and I'll ask you: what's the point of checking the info before confirming?!), contact the administrator of the system.

● **To update a contract**, choose the corresponding option and type in the new information.

```
You are about to update [Client Ouane]'s contract.
Please confirm (y/n):
y
What would you like to update?
        1: Contract's total price
        2: Contract's due amount
        3: Contract's signature
Choice: 3
        The contract has been signed (y/n): y_
```

If the contract has been signed, option 3 will disappear and be displayed as:

```
What would you like to update?
        1: Contract's total price
        2: Contract's due amount
        The contract has already been signed
Choice: _
```

● **To display the events (if any) without an assigned support member**, choose the corresponding option. If there aren't any events or they all have assigned support, the list will be empty. Otherwise, the event(s) will be listed:

```
        EVENT WITHOUT AN ASSIGNED SUPPORT MEMBER:

Event ID: 1
        Related to contract ID: 1
        Client: Client Ouane
        Client's contact information: Client Ouane
        Start date: None
        End date: None
        Location: Pompouy-Saint-Poulet
        Attendees: 5
        Notes: Fiesta fiestaaaaaaaaaa!

        Press ENTER to go back to your menu. _
```

● **To assign support to an event**, choose the corresponding option and enter an existing event's ID. The program will check your inputs, so a wrong event ID or a wrong user ID is not a problem as you will be asked to try again.

```
        Event's ID: 3

        ERROR: That ID does not match any events. Please try again.
        Event's ID: 1

You are about to update the event for [Client Ouane]'s contract (ID: 1).
Please confirm (y/n):
y
        ID of the Support member in charge of that contract (0 if undecided): 234234

        ERROR: That ID does not match any support member. Please try again.
        ID of the Support member in charge of that contract (0 if undecided): wef234f

        ERROR: Only digits are allowed.
        ID of the Support member in charge of that contract (0 if undecided): 1

        ERROR: That ID does not match any support member. Please try again.
        ID of the Support member in charge of that contract (0 if undecided): 9
The event's assigned support member was updated successfully.
        Press ENTER to go back to your menu. _
```

Now, if I display all the events without support, this is what I get:

```
        EVENTS WITHOUT AN ASSIGNED SUPPORT MEMBER:

        Press ENTER to go back to your menu.
```

The list is empty, the update worked, time to relax and get that well deserved coffee.

6. Sales Team

```
SALES MENU | WELCOME, Sales1
* * * * * * * * * * * * * * *
        1. Create a client
        2. Update a client you are in charge of
        3. Update the contract of one of your clients
        4. Display contracts that haven't been signed yet
        5. Display contracts that haven't been fully paid
        6. Create an event (for a client whose contract was signed)
        7. Display all the clients
        8. Display all the contracts
        9. Display all the events

        disconnect. DISCONNECT and go back to authentication
Your choice:
```

You may not be a manager, but you do have quite the powers! Without you, no clients in the database!

To create a client, choose the according option, confirm your password and fill in the fields.

```
CREATION OF A CLIENT:
        Full name: John Dough
        email (optional, press ENTER if no email):
        Telephone number:

        This field is required, please type in something.

        Telephone number: 5411223543
        Name of the Company: Portopullo

        Current date and time: 2023-08-13 11:47:43
        Use the current date? (y/n)? y

Please confirm the new client's details:
        Full name: John Dough
        email: no_email@8d8dc620dca4edfe
        Phone number: 5411223543
        Company name: Portopullo
        Last contacted: 2023-08-13 11:47:49

Confirm? (y/n)
y_
```

To update a client your are in charge of (which means you **cannot** update other salesmen's clients), choose the corresponding option and modify whatever you want.

If the client does not exist or is not "yours", a message will let you know. If things go well, yu will be asked to confirm that you want to update that client.

```
        Client's ID: 2
You are not in charge of that client, please try again with another one.
        Client's ID: 6

        ERROR: That ID does not match any clients. Please try again.
        Client's ID: 4

You are about to update the client 'John Dough'.
Please confirm (y/n):
```

```
What would you like to update?
        1: Client's name
        2: email
        3: Phone number
        4: Company's name
        5: Last contact date
        salesman: Salesman in charge
Choice:
```

Important It is possible to change the salesman in charge (which at the moment is you). If you ever chose to do so, please understand that you will not be able to modify that client's information as it will not be "yours" anymore.

```
Current Salesman ID in charge: 6
        ID of the new Salesman in charge of that contract: 7
The client was updated successfully.
        Press ENTER to go back to your menu. _
```

I changed the salesman in charge of that client. Now if I try to modify their information, this is what I will get:

```
        Client's ID: 4
You are not in charge of that client, please try again with another one.
        Client's ID:
```

● **To update the contract of one of your clients**, choose the corresponding option and type in the new information.

```
You are about to update [Client Ouane]'s contract.
Please confirm (y/n):
y
What would you like to update?
        1: Contract's total price
        2: Contract's due amount
        3: Contract's signature
Choice: 3
        The contract has been signed (y/n): y_
```

If the contract has been signed, option 3 will disappear and be displayed as:

```
What would you like to update?
        1: Contract's total price
        2: Contract's due amount
        The contract has already been signed
Choice: _
```

● **To display contracts that haven't been signed yet or haven't been fully paid**, choose the corresponding option. Note that it will show you the contracts regardless of the salesman in charge.

● **To create an event (for the contract of a client you are in charge of)**, choose the corresponding option and type in the new information. Keep in mind that the contract must have been signed, or else it won't be possible:

```
CREATION OF AN EVENT:
        Contract's ID: 4

        ERROR: The contract you referred to has NOT been signed yet.
        Press ENTER to go back to your menu.
```

Upon entering a right ID, you will be asked to filli in the required fields.

```
CREATION OF AN EVENT:
        Contract's ID: 3
You are not in charge of that client, please try again with another one.
        Contract's ID: 2
        Client's contact info (name, phone number, etc.): Mister Shlouboudou, 0255448876, Old York Village
        Start date of the event (press ENTER to skip that step)
        Year:
        End date of the event (press ENTER to skip that step)
        Year:
        ID of the Support member in charge of that contract (0 if undecided): 0
        Location: Billu Island
        How many attendees? 16
        Additional notes: Date to be decided

Please confirm the new event's details:
        Contract's ID: 2
        Client: Client Ouane
        Client's contact information: Mister Shlouboudou, 0255448876, Old York Village
        Starts on: None
        Ends: None
        Support member's ID: None
        Location: Billu Island
        Number of attendees: 16
        Notes: Date to be decided

Confirm? (y/n)
```

You can then check that the event was successfully saved into the database by displaying all the events (in my case, Event ID 2):

```
        LIST OF ALL THE EVENTS:

Event ID: 1
        Related to contract ID: 1
        Client: Client Ouane
        Client's contact information: Client Ouane
        Start date: None
        End date: None
        Support member in charge: Support Tou (ID: 9)
        Location: Pompouy-Saint-Poulet
        Attendees: 5
        Notes: Fiesta fiestaaaaaaaaaa!

Event ID: 2
        Related to contract ID: 2
        Client: Client Ouane
        Client's contact information: Mister Shlouboudou, 0255448876, Old York Village
        Start date: None
        End date: None
        Support member in charge: not assigned yet
        Location: Billu Island
        Attendees: 16
        Notes: Date to be decided

        Press ENTER to go back to your menu.
```

7. Support Team

```
SUPPORT MENU | WELCOME, Support1
* * * * * * * * * * * * * * *
        1. Update an event you are in charge of
        2. Display the events you are in charge of
        3. Display all the clients
        4. Display all the contracts
        5. Display all the events

        disconnect. DISCONNECT and go back to authentication
Your choice:
```

Yes, there is not much you can do, but without you, no party can take place!

● **To update and event**, choose the corresponding option. It will not work if you don't enter an event you are in charge of. If you are supposed to be in charge of an event and it does not work, contact your manager, as he/she is the one who can affect support to events.


```
You are about to update the event for [Client Ouane]'s contract (ID: 2).
Please confirm (y/n):
y
What would you like to update?
        1: Client's contact info
        2: Start date
        3: End date
        4: Assigned Support member
        5: Location
        6: Number of attendees
        7: Notes
Choice:
```

Important It is possible to change the support member in charge (which at the moment is you). If you ever chose to do so, please understand that you will not be able to modify that event's information as it will not be "yours" anymore.

Through the menu, choose whatever you need to update.

About Start date and end date:
The dates are checked inside the program to prevent any mistakes or fraud attempt. As such, the end date cannot be prior to the start date. The start date cannot be prior to the current date. In case you made a mistake, contact the administrator.
For example (the current date is 2023/08/13):

```
Choice: 2
        Start date of the event (press ENTER to skip that step)
        Year: 2023
        Month: 8
        Day: 12
        Hours: 15
        Minutes: 30
The start date cannot be prior to the current date.
        Start date of the event (press ENTER to skip that step)
        Year:
```

But if I do:

```
        Start date of the event (press ENTER to skip that step)
        Year: 2023
        Month: 8
        Day: 14
        Hours: 18
        Minutes: 30
Is that date correct?
        2023-08-14 18:30:00 (y/n): y
The event was updated successfully.
```

It worked, as the chosen date is in the (relative) future.

Same goes for the end date:

```
Choice: 3
        Event's start date: 2023-08-14 18:30:00.
        End date of the event (press ENTER to skip that step)
        Year: 2023
        Month: 8
        Day: 14
        Hours: 18
        Minutes: 20
The end date cannot be prior to the start date.
        Event's start date: 2023-08-14 18:30:00.
        End date of the event (press ENTER to skip that step)
        Year: _
```


But:

```
Is that date correct?
        2023-08-14 20:30:00 (y/n):
```


And as we check in the events list (or the events you are assigned to):

```
Event ID: 2
        Related to contract ID: 2
        Client: Client Ouane
        Client's contact information: Mister Shlouboudou, 0255448876, Old York Village
        Start date: 2023-08-14 18:30:00
        End date: 2023-08-14 20:30:00
        Support member in charge: Support Ouane (ID: 8)
        Location: Billy Island
```

MANAGEMENT TEAM

1. Create a user
2. Update a user
3. Delete a user
4. Create a contract
5. Update a contract
6. Display events without assigned support
7. Assign support to an event (= update event)
8. Display all the clients
9. Display all the contracts
10. Display all the events

SALES TEAM

1. Create a client
2. Update a client you are in charge of
3. Update the contract of one of your clients
4. Display contracts that haven't been signed yet
5. Display contracts that haven't been fully paid
6. Create an event (for a client whose contract was signed)
7. Display all the clients
8. Display all the contracts
9. Display all the events

SUPPORT TEAM

1. Update an event your are in charge of
2. Display the events you are in charge of
3. Display all the clients
4. Display all the contracts
5. Display all the events

COMMON TO ALL TEAMS

Display all the clients
Display all the contracts
Display all the events