created the **repo on git**installed the fastapi , sqlalchemy , psycopg2, alembic , pydantic  
created the **schema and model of user**  
after that uses command  
 **- > alembic init alembic**   
make changes in the line **21 of env.py file**   
**config.set\_section\_option('alembic','sqlalchemy.url',f'postgresql+psycopg2://postgres:{postgresSQL\_password}@localhost:5432/{database\_name}')  
from model import Base  
target\_metadata = Base.metadata**  
after that   
 **- > alembic revision --autogenerate -m "init user model"**check version file (in that contain the user models and columns names )  
after that uses command  
 **- > alembic upgrade head**

it will create the user table in the database .

JWT tokens and SMTP :

Json web tokens : user id as a data and the SH256 algorithm and 32 bit hexadecimal secret key to encode the data using JWT . using this we create the token and send the user registered email using

simple mail transfer protocol we send the mail with verification link is link the (http://127.0.0.1:8000/user/verify?token=”asdfafa65a46dfa5s6df4a5fd6a5dfa6fd5a4asd”)

creating secret key : **import secrets > secrets.token\_hex(32)**

after sending the mail to the user . then user can hit that link then the verify api is automatically called . in the **verify api** we make the **is**\_**verified** filled In the database as the **Trur**

add the notes model and schema also the in notes.py added the notes apis in the add update delete and get the notes .

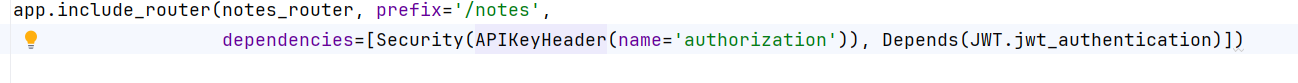
add the logging in the notes and user apis

* **alembic revision --autogenerate -m "init notes models"**
* **alembic upgrade head**

/APIRouter :

**Authentication** is verifying the who someone is , whereas

**Authorization** is the process of verifying what specific applications , files and data a user has access to.



**app.include\_router**: This is a method provided by FastAPI to include a router in your application. It is used to modularize and organize your API endpoints. (used for make the big application )

**dependencies**: This parameter is used to specify any dependencies that should be applied to all routes in the included router.

**Security(APIKeyHeader(name='authorization'))**: This is using FastAPI's **Security** dependency to enforce API key security. It expects an API key to be included in the header with the name 'authorization'. If the API key is not present or incorrect, it may raise an exception or handle it based on your configuration.

4 Labels APIS :

Add the label apis and the create the model and schema for that

> alembic revision --autogenerate -m "init labels models"

> alembic upgrade head

5 Redis :

Redis use the ram memory of the system (cache memory )

For **windows** we uses the **memurai as a redis server**

**Redis :** <https://winget.run/pkg/Memurai/MemuraiDeveloper>

<https://winget.run/pkg/Memurai/MemuraiDeveloper> <https://redis.io/docs/connect/clients/python/> <https://www.youtube.com/watch?v=R4Y0TBrEAak> <https://www.youtube.com/watch?v=_8lJ5lp8P0U> <https://www.youtube.com/watch?v=XCsS_NVAa1g>

so we are adding the redis in the notes apis

hset(key, field, value) , hget(key,field) her is **key is user\_id** and **field is the note id**

6 Celery :

We use that for sending the mail asynchronously . celery having the broker and workers and it preform the task .

*# running the celery workers   
#* ***celery -A task.celery worker -l info --pool=solo -E***

7 Middleware :

We adding the middleware for counting the how many time requested on that path we are counting