

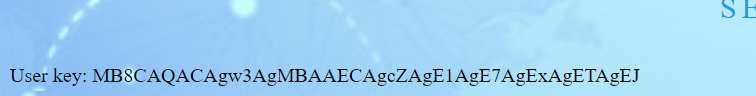
|  |  |  |  |
| --- | --- | --- | --- |
| Marwan Tarek | Sebastiano Saad | Mohamed Halfawy | Mohamed Magdy |
| 18101712 | 18101886 | 18101742 | 18100322 |
| UI | Uploading file | Encryption, decryption | database |

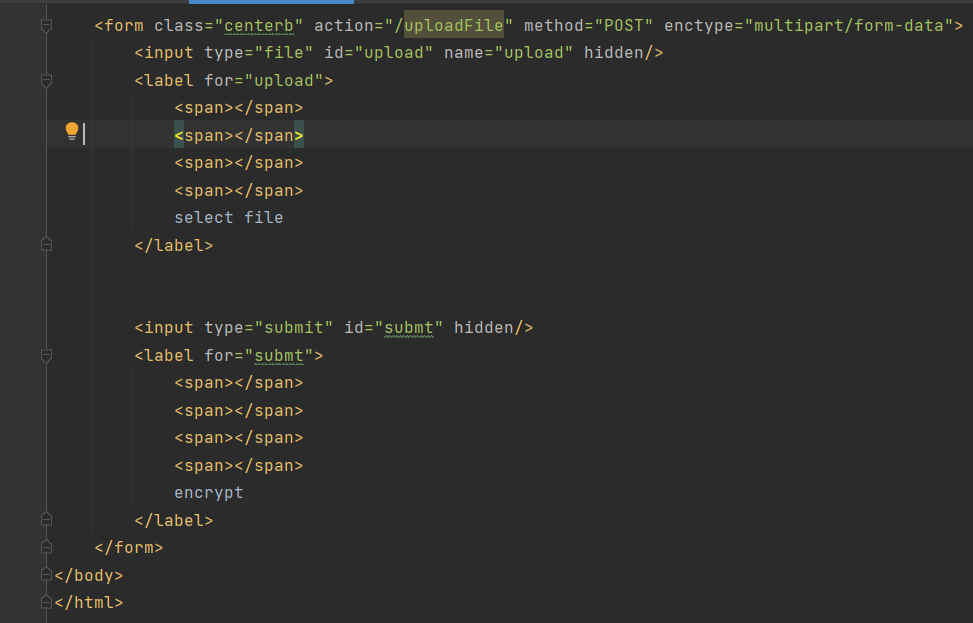
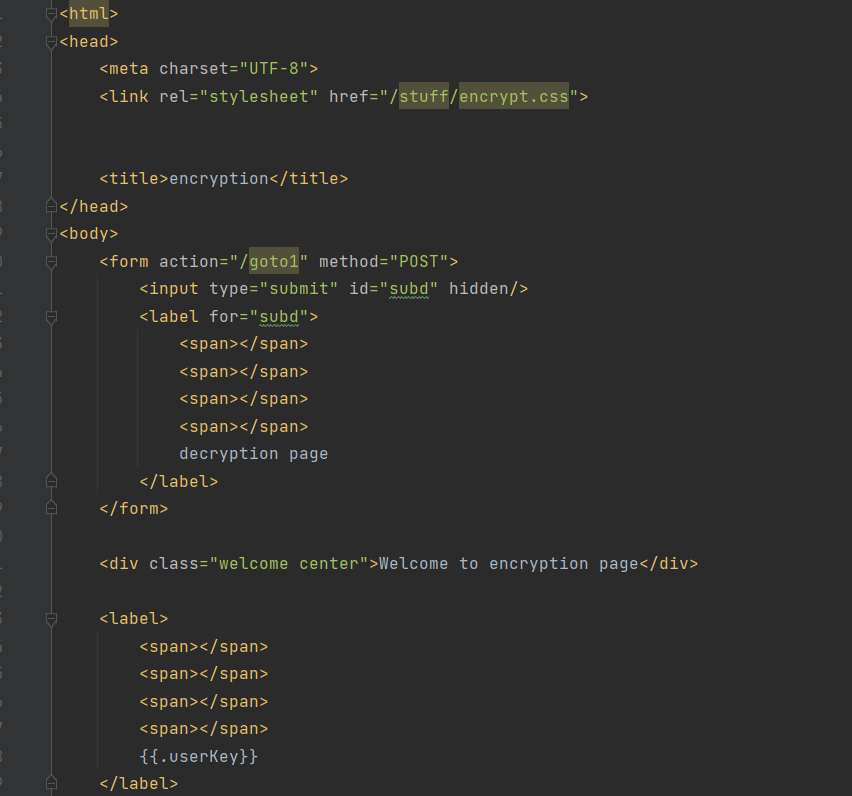
Brief idea of the project

The purpose of the project is where the user can Encrypt his own preferable file and we are responsible for encrypting and saving the encrypted file and we generate a unique key given to the user to decrypt his file when desirable then on our end we decrypt the when user chooses to decrypt the file we use this key ,which is linked another key generated for further safety only visible at backend then decryption phase takes place and file is available again.

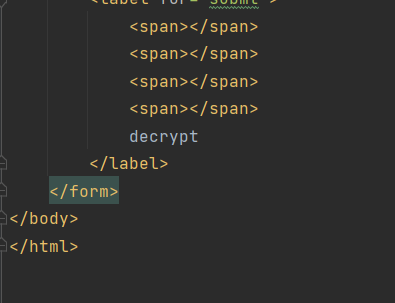
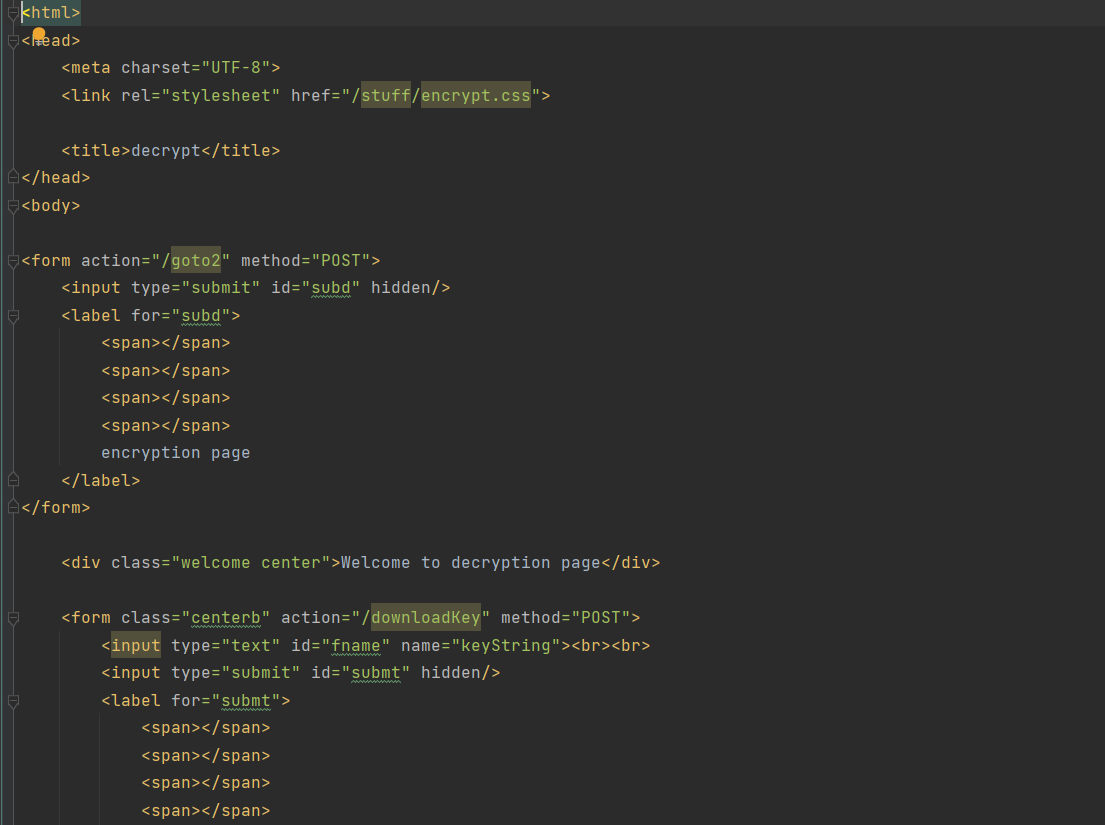
Screenshots of run and code



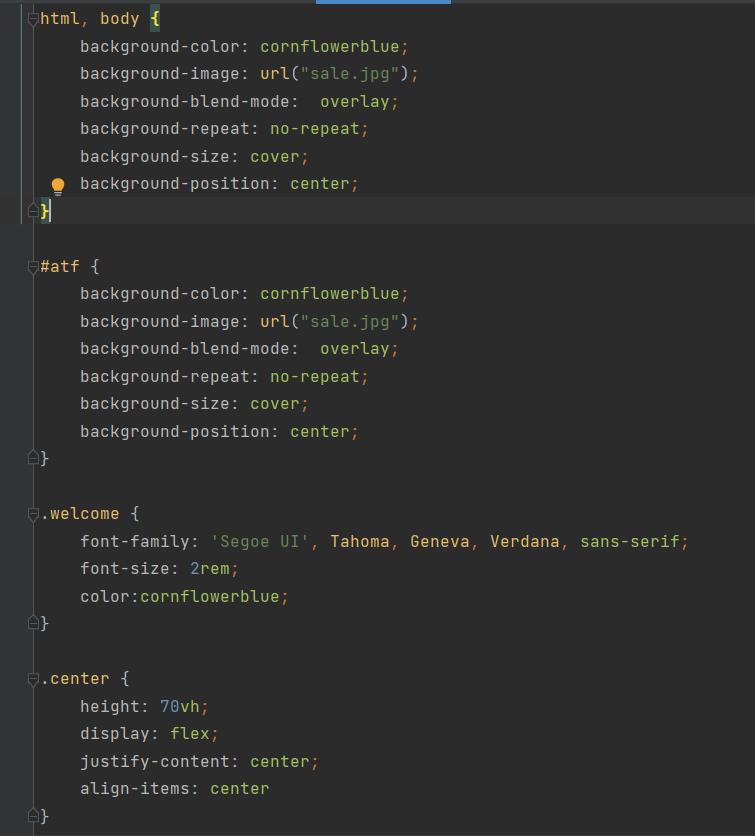
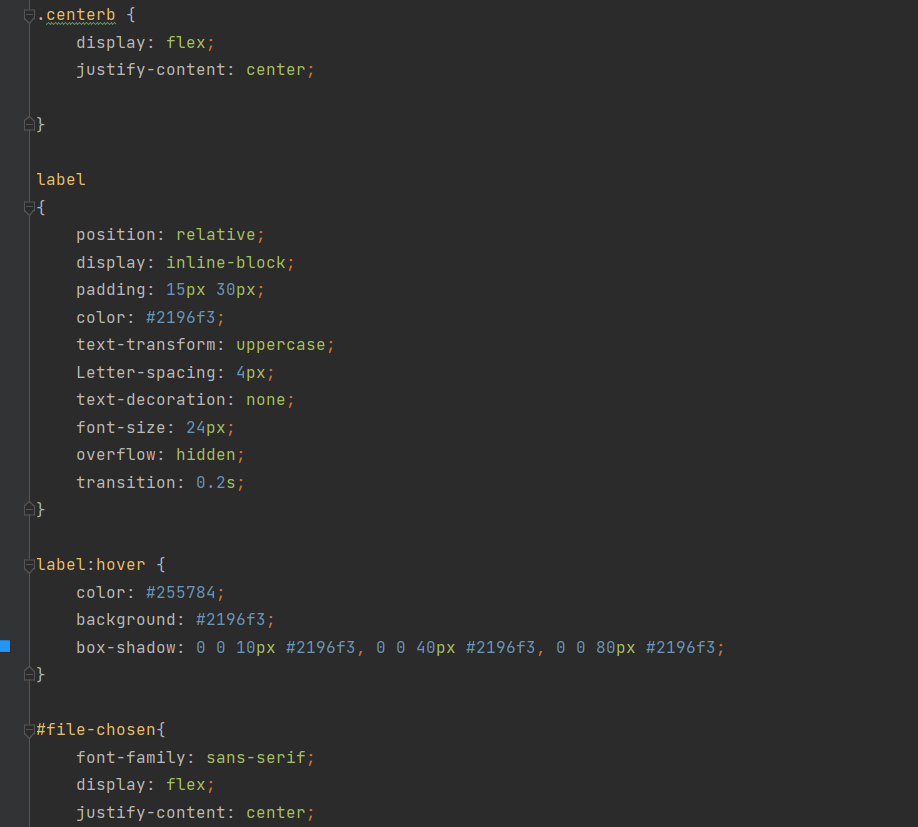
-firstly, after run user is met with the encryption page where there is a Decryption page button at the top which will redirect user to decryption page ,to add more there is a select file option(button) where user is given free option to choose his file ,followed by an encrypt button when pressed, the file will get encypted and a user key will get displayed for user to get copied(later used for dectyption) User key generated



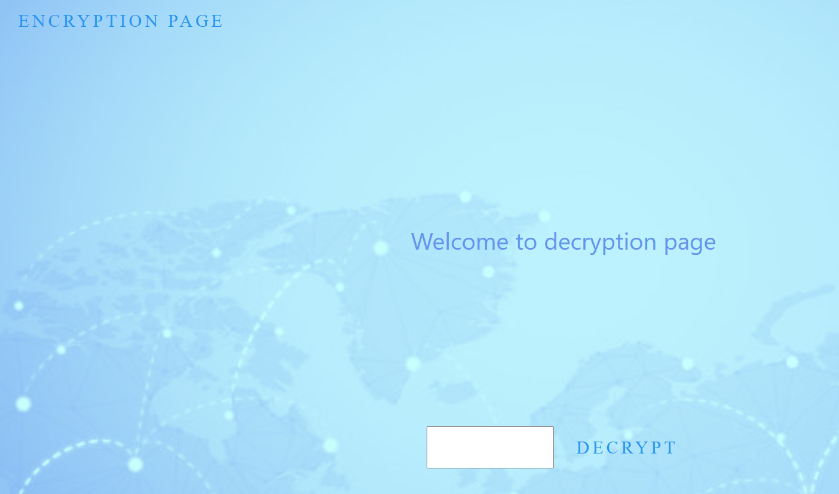
Html for encryption page



Html for decryption page



Css for encryption and decryption page



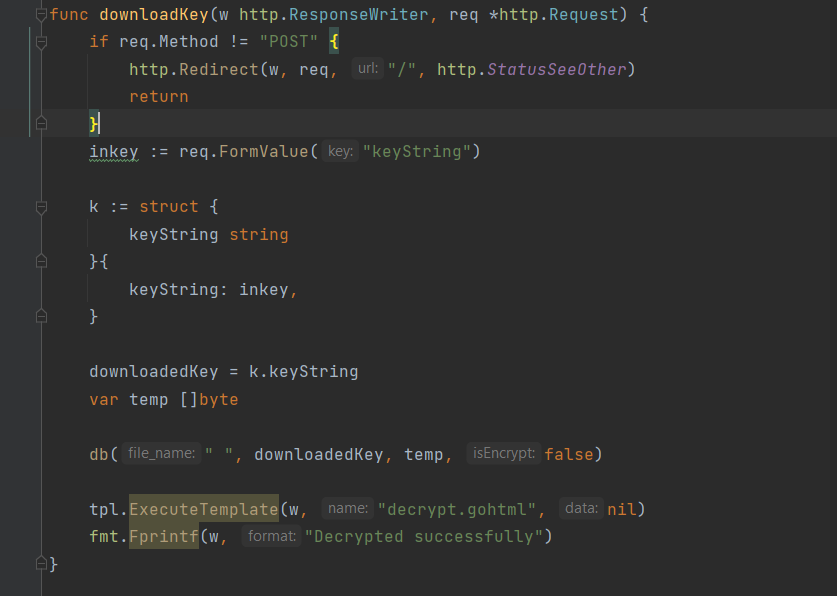
---> if decryption button pressed user will get redirected to decryption page where he enters his key and then followed by decrypt button that will display decryption successful and this button is connected to the code which will send the user key and sent to the database to get the file key connected to the user key and with this key it is send to the decrypt file function which is responsible for decrypting the file



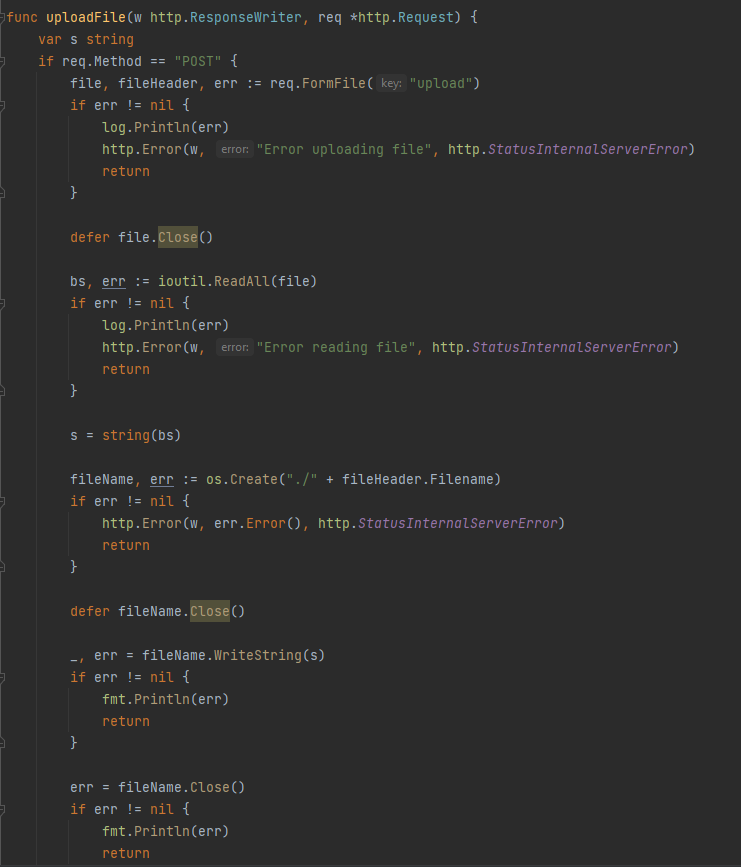
* This is the main function responsible for calling the function to be executed.
* And initialize the local host.

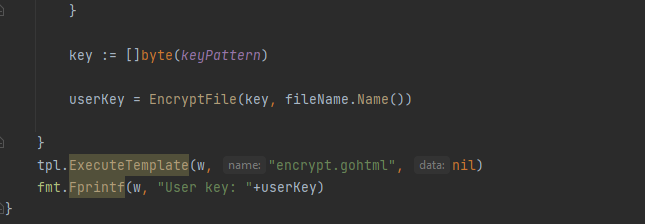


* Decr function responsible for viewing the ui of decrypt page.
* Encr function responsible for viewing the ui of encrypt page.



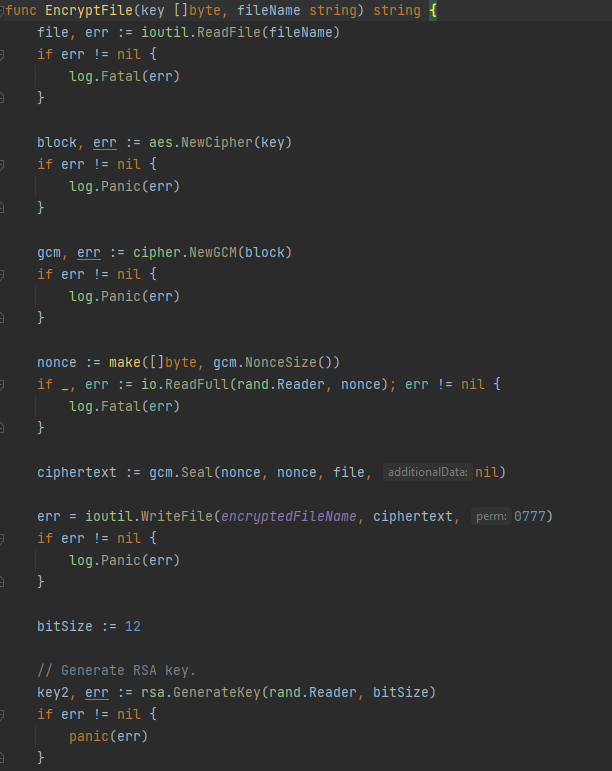
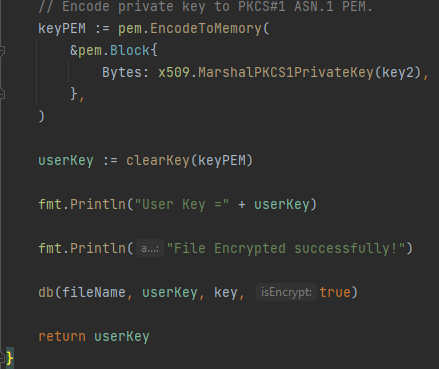
* This function we fetch the key string from the user.
* And we create a struct to set the value with the string.
* And then call the db function to execute.
* At the end we tell the user that the decryption has been done successfully.





I check if the http Request received is equal POST it will execute the following:

* The user will upload a file and it will be saved in the variable file
* It will read all the file and will save it in the variable “bs”
* Then will save all the file in s variable in strings
* It will create in a file with the same name and add it in the project
* Then it will write all the strings in the file created
* Then I call the function for encryption
* At the end I execute the ui template and show the user key after the encryption

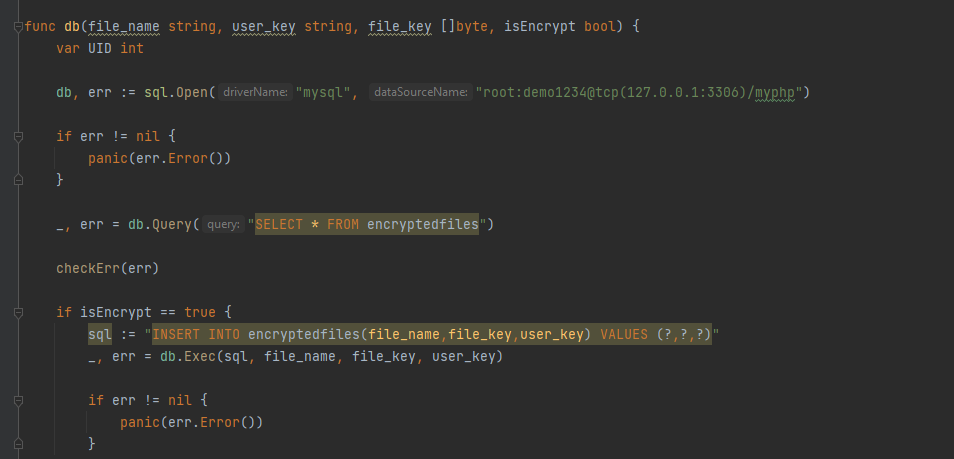
Here i start encrypting the file through the following steps:

* First we read the file the user uploaded from the website
* Then we crate the block for our algorithm where the file is encrypted using fixed-size blocks from the key we generated
* Then we use the GCM mode which is a stream mode that encrypts the file with every single character inside of it with authentication
* The GCM mode requires nonce array where it puts all the key in an array of bytes
* Then we encrypt the data by using the seal function it encrypts the file using GCM mode and appending the nonce and tag to the final data so we can use it to decrypt the file later
* Then lastly we save our cyphertext into our project
* Next we create the key that we give to the user so we can get the file key from the database when we need to decrypt it
* We create it a key of size 12 bits then use the package RSA which generate a RSA key by passing the size of the bits of the key and random numbers/characters to create it
* Then we encode the private key to PEM and save the key in keyPEM variable
* Then we clear all the nonused characters in the key String and finally send it to the database



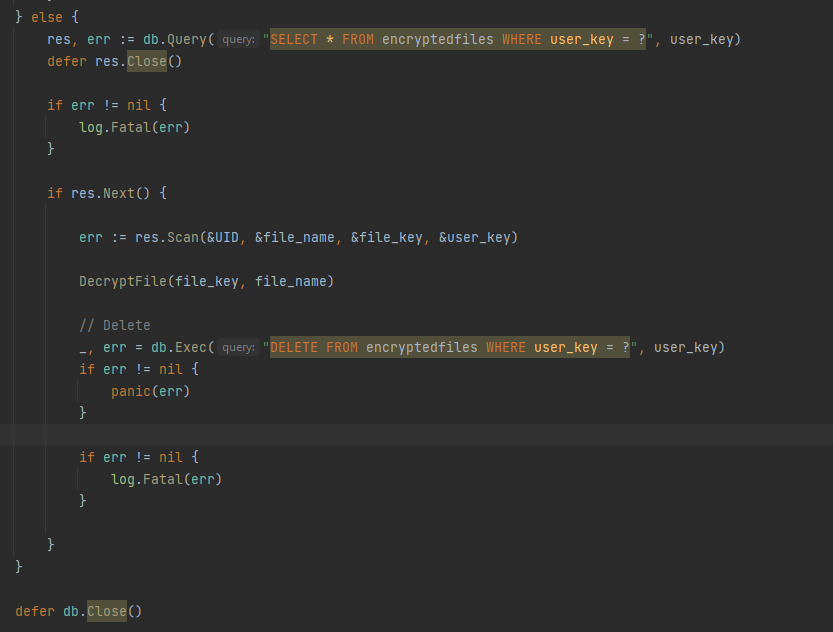
Here we decrypt the file to get it back to the user:

* To decrypt the file first we read the encryptedFile.bin which contains all the keys of the files
* Then we repeat the same process above we create a block from the key and and use the GCM for stream mode with authentication and nonce to using the GCM mode
* Lastly we use the gcm.Open function and pass the nonce value we used in the encryption process to get the same file the user encrypted
* And finally we adjust the filename and extension to save the file in the project as the user encrypted it



Function database is responsible for saving both user key that is generated for the user and the file key generated only for backend purposes of usage of decrypting the file and file name

* The function receives the file name , file key, user key , and a bool that is responsible for either running the encryption part of code or decryption part
* Firstly a database connection is established and if not connected a panic takes place
* If bool == true encryption part takes place where file name and file key and user key is saved in a table(encrypted files) -information stored in sequence -if error occurs panic takes place



* If bool is false decryption takes place and a scan through the whole table takes place to search with user key that is taken from user and if found we retrieve the file information and we send the file key with the file name to DecryptFile function so decryption takes place
* After decryption , the decrypted file is not needed so descareded from database
* Finally we close the connection with database

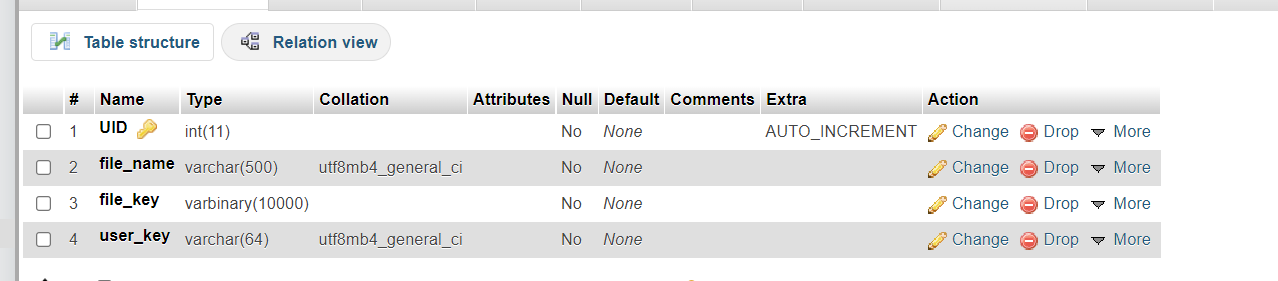
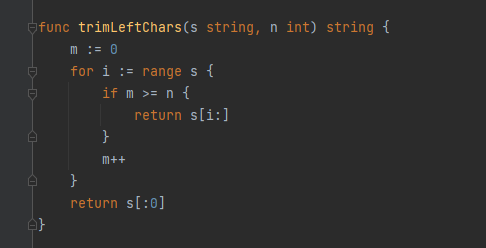


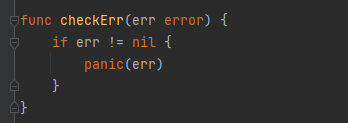
Table structure



-filter of encryption key from any additional spaces or strings



-trim most left character from string



Function to check error

RUN:



