How does the code work?

This code implements a basic encryption and decryption scheme using the Advanced Encryption Standard (AES) block cipher.

The user is prompted to choose an operation, either encryption or decryption. They are also asked to provide the file path and a secret key for encryption or decryption.

The code then reads the contents of the file specified by the file path and stores the content in a byte string called file_bytes.

Next, the code creates an AES cipher object with the specified key and an initialization vector (IV) for encryption or decryption, depending on the user's choice of operation. The cipher object is created using the AES.new() method from the Crypto.Cipher library.

If the user chooses encryption, the encrypt() method is called on the cipher object to encrypt the contents of file_bytes. Otherwise, if the user chooses decryption, the decrypt() method is called on the cipher object to decrypt the contents of file_bytes.

The resulting encrypted or decrypted byte string is stored in new_file_bytes. Finally, the code writes the new byte string to the same file specified by file_path.

Note that the IV used for encryption and decryption must be the same, and it should be unique for each message to prevent attacks. In this code, the IV is set to "kaasfasf", but it is recommended to use a cryptographically secure random value.

How does AES work?

AES (Advanced Encryption Standard) is a symmetric encryption algorithm that works by dividing the input data into fixed-size blocks and then applying a series of substitution, permutation, and mixing operations.

AES uses a secret key to perform these operations, which can be 128, 192, or 256 bits long. The key is used to generate a set of round keys that are used in each round of the encryption process.

During encryption, each block of data undergoes several rounds of transformation, where the block is mixed with the round key using a combination of substitution and permutation operations. The result of each round is then used as input to the next round until the final round, which produces the ciphertext.

Decryption works by applying the inverse operations in reverse order, starting with the final round and working backwards. Only someone with the secret key can decrypt the ciphertext and retrieve the original plaintext.