

# FILE HASHER

Alright folks, gather around because today we're going to learn how to create a simple file hasher in python code. And who doesn't love hashing their files? It's like putting your files on a healthy, gluten-free, low-carb diet.

So, what exactly is a file hash, you might ask? A file hash is a unique representation of the contents of a file. Think of it like a digital fingerprint for your file. It's a way to ensure that the file hasn't been tampered with or changed in any way.

Let's start by importing the required libraries. We'll be using the hashlib library in python, which provides various hash functions.

## Python Code

```
import hashlib
```

Next, we'll define a function to calculate the hash of a file. This function will take in the file path as an argument and return the hexadecimal representation of the hash.

## Python Code

```
def hash_file(file_path):
    BUF_SIZE = 65536
    hasher = hashlib.sha256()
    with open(file_path, 'rb') as file:
        buf = file.read(BUF_SIZE)
        while len(buf) > 0:
            hasher.update(buf)
            buf = file.read(BUF_SIZE)
    return hasher.hexdigest()
```

In the function above, we first create an instance of the sha256 hash function. Then, we open the file using the **with** statement and read it in chunks of size **BUF\_SIZE**. We update the hash with

each chunk of the file, and when the entire file has been read, we return the hexadecimal representation of the hash using the **hexdigest** method.

Finally, we can use this function to hash any file and compare the hashes to ensure that the contents of the file haven't been changed. For example, let's hash a file located at '**file.txt**':

#### Python Code

```
file_path = 'file.txt'  
print(f'Hash of {file_path}: {hash_file(file_path)}')
```

And that's it! You now have a simple file hasher in python code. Just be careful not to accidentally hash your grandma file, that could be a disaster.

I hope you've enjoyed this tutorial, and if you've made it this far, congratulations, you're one step closer to becoming a professional file hasher!

