1.how to use multer library in node



Ans:

Step 1: Run this command on your project terminal. By running this the multer library install in your project .

npm install multer

Step 2: Like the image below, we need to import the library . where you want to use the multer in your project you need to import in that file .

```
刘 File Edit Selection View Go Run …

∠ Local-Style-Spot-API

        EXPLORER
                                                                                                             JS products.is M X
Ф

∨ LOCAL-STYLE-SPOT-API

                                               [+ F ひ 🗗 Style-Spot-API > controller > JS products.js > 📵 addProduct > 📵 result

✓ Style-Spot-API

                                                                            const pool = require('../config/database')

✓ confia

          JS database.js
                                                                            const ds= multer.diskStorage({
         controller
                                                                                 destination:'product-uploads
filename:(req, file, cb)=>{
Js products.js
                                                                                     cb(null,Date.now()+file.originalname)
            ≡ signup
                                                                                 storage:ds
                                                                    const addProduct = async (req.res) =>{
console.log('----->>>product req',req.body)
console.log('---->>>product_file_path',req.file)
try{
          gitignore
         JS app.js
         {} package-lock.json
                                                                    const {product_name,product_price,product_category,product_color,product_description,user_id}
console.log('----->>>product_name, product_price, product_category, product_color, product_de
const product_file_path = req.file ? req.file.path : null;
         {} package.json

    README.md

        {} package-lock.json
                                                                    PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
                                                                                                                                                                                            ☑ powershell + ∨ Ⅲ 🛍
                                                                  O PS C:\Users\chettipally\Desktop\Local-Style-Spot-API>
      > OUTLINE
    $P suri-style-spot-BE* ← ⊗ 0 🛦 0 🐕 0
                                                                                                                                                               Ln 25, Col 134 Spaces: 4 UTF-8 CRLF () JavaScript @
```

```
const multer = require('multer')
To use it like above, we need to import it to our folder.
const ds= multer.diskStorage({
    destination:'product-uploads/',
    filename:(req, file, cb)=>{
        cb(null,Date.now()+file.originalname)
    }
})
const upload = multer({
    storage:ds
})
```

The above logic handles the file it is coming from the front end .In the code destination: 'product-uploads/', creates the folder if no folder in our project .this is the (the created one) folder it stores all the files that are coming from the front end.

```
filename: (req, file, cb) => {
    cb(null, Date.now()+file.originalname)
}
```

This code is to save the name of the file .It saves the file name with today's date .

Step 3: in route.js we need like below here we calling the function actually before the api

```
∨ LOCAL-STYLE-SPOT-API

                                                              Style-Spot-API > J5 router.js > [∅] express
                                                               const pool = require('./config/database')
const express = require('express')

✓ Style-Spot-API

✓ config

                                                                     const app = express()
         JS database.js
                                                                4 const {getstudentinfo} = require('./controller/student')
                                                                     const {signupuser,signinuser} = require('./controller/signup')
                                                              const {addProduct, getproduct, upload} = require('./controller/products')
app.get('/studentinfo', getstudentinfo);
         JS products.is
          ≡ signup
          JS student.js
                                                               9 //seller API'S
10 app.post('/signupuser', signupuser)
                                                               11 app.post('/signinuser', signinuser)
🖬 1722513821755desadvantages.png
        .env
                                                               app.post('/add-product', upload.single('product_file_path'),addProduct)
app.get('/getAllproducts',getproduct)
         gitignore
         JS app.js
        {} package-lock.ison
        {} package.json

    README.md

        JS router.is
       {} package-lock.json
```

```
app.post('/add-product', upload.single('product_file_path'),addProduct)
```

Why can't we save the file that is coming from the frontend directly in the database?

ANS: it is really not a good idea to save the file in the database.

- 1.Efficiency: Databases are optimized for handling structured data, not large binary files. Storing large files in the database can degrade performance or efficiency.
- 2.Query Speed: Retrieving and manipulating binary data in a database can be slow compared to accessing file paths and loading files from the filesystem.

Because of the above reasons we use the multer library to handle the files better  $\boldsymbol{.}$ 

## \*\*\*IMP NOTE:

With the multer library we are writing the middleware function to handle the files. Where it creates the folder and stores the files.

Now we are inserting the file address in the database, not the storing the file in the database.

This is the good approach to handle the files