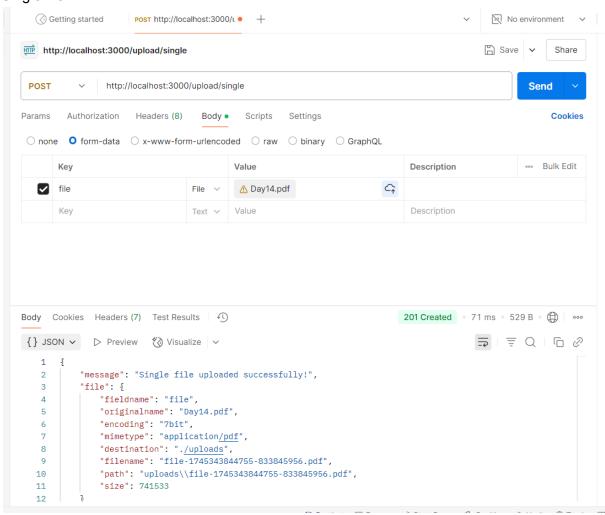
```
import {
    Controller,
    Post,
   UploadedFile,
    UploadedFiles,
    UseInterceptors,
  } from '@nestjs/common';
  import { FileInterceptor, FilesInterceptor }
from '@nestjs/platform-express';
  import { diskStorage } from 'multer';
 import { extname } from 'path';
 @Controller('upload')
 export class FileUploadController {
    @Post('single')
    @UseInterceptors(FileInterceptor('file', {
      storage: diskStorage({
        destination: './uploads',
        filename: (req, file, cb) => {
          const uniqueSuffix = Date.now() + '-' +
Math.round(Math.random() * 1e9);
          cb (null,
${file.fieldname}-${uniqueSuffix}${extname(file.o
riginalname) } `);
       },
      }),
    })))
```

```
uploadSingle(@UploadedFile() file:
Express.Multer.File) {
      return {
       message: 'Single file uploaded
successfully!',
       file,
      };
   @Post('multiple')
    @UseInterceptors(FilesInterceptor('files', 5,
     storage: diskStorage({
       destination: './uploads',
       filename: (req, file, cb) => {
         const uniqueSuffix = Date.now() + '-' +
Math.round(Math.random() * 1e9);
         cb (null,
${file.fieldname}-${uniqueSuffix}${extname(file.o
},
      }),
    }))
   uploadMultiple(@UploadedFiles() files:
Express.Multer.File[]) {
     return {
       message: 'Multiple files uploaded
successfully!',
```

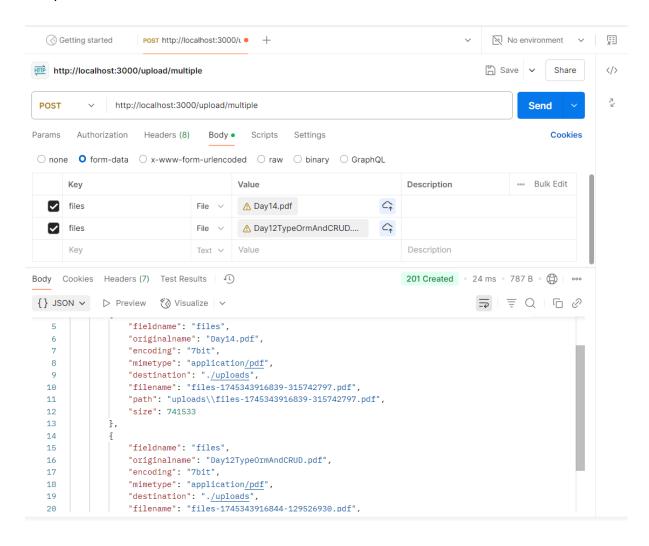
```
files,
};
}
```

OUTPUT

Single file



Multiple files



Buffer

```
import {
    Controller,
    Post,
    UploadedFile,
    UseInterceptors,
} from '@nestjs/common';
import { FileInterceptor } from
'@nestjs/platform-express';
```

```
import * as multer from 'multer';
 @Controller('buffer')
 export class BufferUploadController {
    @Post('upload')
    @UseInterceptors(FileInterceptor('file', {
      storage: multer.memoryStorage()
    }))
    handleFileUpload(@UploadedFile() file:
Express.Multer.File) {
      return {
        message: 'File uploaded to buffer
successfully!',
        filename: file.originalname,
        mimetype: file.mimetype,
        sizeInBytes: file.buffer.length,
        preview:
file.buffer.toString('base64').substring(0, 100) +
      };
```

