

1.npm init -y 初始化

2.下载express

npm i express -save

3.引入

```
const express=require('express')
```

```
const app=express()
```

```
1
```

4.自动运行命令

去pakg.jsion中添加 脚本服务

```
test : echo \ Error: no test specified \ && exit 1
"dev": "nodemon servse.js"
},
```

expressGET请求

```
1  const express=require('express')
2  app=express()
3
4  app.get('/', (req, res) => {
5    res.send('返回get的响应')
6  })
7  app.listen(3000,()=>{
8    console.log(3000);
9  })
```

post请求

设置post的中间件

```
1  app.use(express.json()) //json 格式
2  app.use(express.urlencoded({ extended: true }))) //原生表单
```

如果是formdata格式

下载 formidable()

```
1  npm install formidable --save
```

引入

```
1 const formidable = require('formidable')
```

看文档使用

```
1 app.post('/upload', (req, res) => {
2   const form = formidable({});
3   form.parse(req, (error, fields, files) => {
4     if (error) {
5       return;
6     }
7     res.json({fields, files})
8     console.log(fields,files);
9
10  })
11 })
```

post请求

```
1 app.post('/books', (req,res)=> {
2   console.log(req.body)
3   res.send('post')
4 })
5
```

express 中间件

中间件函数能够访问[请求对象](#)（`req`）、[响应对象](#)（`res`）以及应用程序的请求/响应循环中的下一个中间件函数。下一个中间件函数通常由名为 `next` 的变量来表示。

通过`text ()` 交给下一个中间件处理

错误处理中间件

```
1 app.use(function(err, req, res, next) {
2   console.error(err.stack);
3   res.status(500).send('Something broke!');
4 });
```

引入日志中间件可以查询日志

```
1 npm i morgan --save //安装
2 const morgan=require('morgan')
3
```