# 打开上一节我们创建的express\_hello项目，我们继续来学习

|  |
| --- |
|  |

## 其实你可以在express应用程序里面写nodejs原生代码，没有冲突，只是express有一个静态fwq中间件static，它可以简化我们的操作，我们再也不用像以前那样子写了，直接app.use(express.static(静态资源服务器文件夹就))就ok了

# 我们做一个get实例，还是用上面的姓名

## 1）在public目录下面新建一个register.html,内容如下

|  |
| --- |
| <!DOCTYPE html>  <html lang="en">  <head>      <meta charset="UTF-8">      <meta name="viewport" content="width=device-width, initial-scale=1.0">      <title>User Register</title>  </head>  <body>      <h3>用户注册</h3>      <form action="/get\_reg" method="get">          <p>用户姓名：<input type="text" name="username" placeholder="输入用户名"></p>          <p>用户邮箱：<input type="text" name="email" placeholder="输入邮箱"></p>          <p>用户密码：<input type="text" name="password"></p>          <p>确认密码：<input type="text" name="confirm"></p>          <p><input type="submit" value="注册"></p>      </form>  </body>  </html> |

### 注意：这里使用get方法只是用来测试express对get请求的处理能力，实际上用户注册一个用post方法

## 2.在02-server-get.js里面添加对/get\_reg的路由处理

|  |
| --- |
| //express处理get请求  let express = require('express')  let app = express()  let PORT = 3001  app.get('/',(req,res)=>{      res.send('<h1>Welcome to index</h1>')  })  app.get('/get\_reg',(req,res)=>{      console.log(req.query);      res.send('<h1>login</h1>')  })  //配置静态服务器  app.use(express.static('public'))  //配置404  app.get('\*',(req,res)=>{ //如果我们没有为两个路径配置路由，就会被他匹配到      res.send('<h1>Not Found</h1>')  })  app.listen(PORT,()=>{      console.log(`url: http://localhost:${PORT}/`);  }) |

## 3.测试在注册页面里面填写内容，然后点击提交按钮，效果如下

|  |
| --- |
|  |
|  |

### 服务器控制台输出如下内容

|  |
| --- |
|  |

### 可见：express可以解析get请求路径后面的查询字符串把它变为一个对象，然后我们可以从这个对象里面获取对应的键的值可以做进一步处理

# 这一节比较简单，下一节我们学习获取post请求的参数