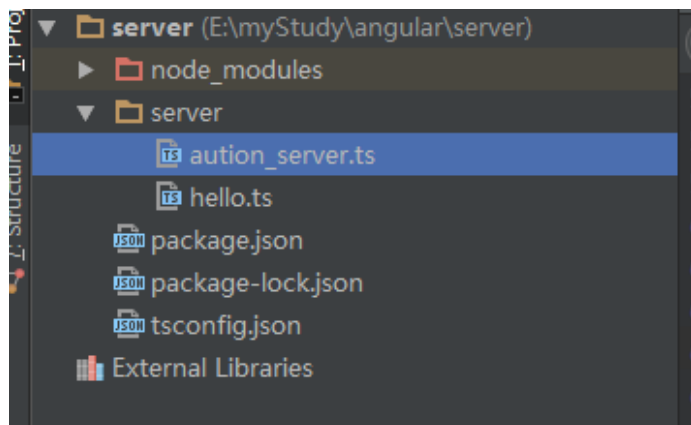


创建Web服务器：

- 1：使用node.js创建服务器
- 2：使用Express创建restful的http服务
- 3：监控服务器文件的变化

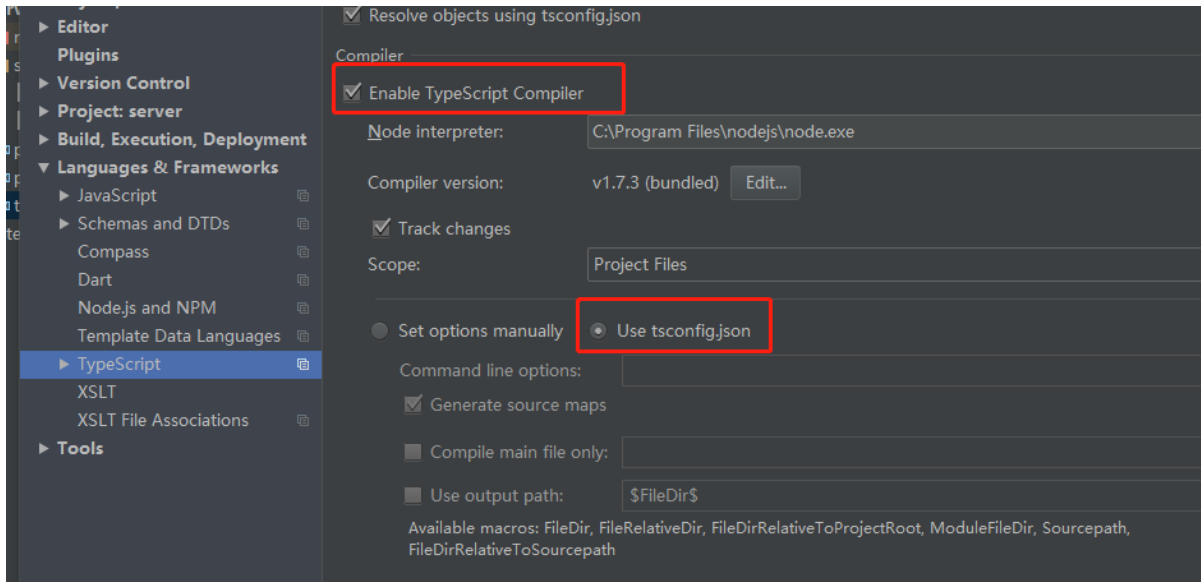


步骤：

- (1) npm init y
- (2) npm i @types/node --save
- (3) 创建tsconfig.ts -----

```
-----{
  "compilerOptions":{
    "target":"es5",
    "module":"commonjs",
    "emitDecoratorMetadata":true,
    "experimentalDecorators":true,
    "outDir":"build",
    "lib":["es6"]
  },
  "exclude":[
    "node_modules"
  ]
}
```

- (4)设置typescript配置



(5) 创建 server/server.ts

```
-----var http = require('http')

const server = http.createServer((request,response)=>{
    response.end('Hello Node!')
})

server.listen(8000);
```

(6)命令行: node server/hello.ts

强化版

(7)npm install express --save

(8)npm install @types/express --save

(9)创建 server/aution_server.ts

```
-----var express = require('express');
const app = express();
const products = [
    { "id":1,
      "title":"第一个商品",
```

```

        "price":1.99,
        "rating":1.5,
        "desc":"这是第一个商品",
        "categories":["电子产品",'硬件设备']
    },
    { "id":2,
      "title":"第二个商品",
      "price":2.99,
      "rating":2.5,
      "desc":"这是第二个商品",
      "categories":["图书"]
    }
  ];
app.get('/',(req,res)=>{
    res.send("Hello Express");
})
//查询products接口的数据
app.get('/products',(req,res)=>{
    res.json( products );
})
//通过products中某件商品的id查询商品
app.get('/product/:id',(req,res)=>{
    res.json( products.find((product)=>product.id == req.params.id))
})
const server = app.listen(8000,'localhost',()=>{
    console.log("服务地址已启动, 地址是: http://localhost:8000")
})

```

(10) npm install -g nodemon //监测内容改变, 自动重启接口, 刷新页面就好了

(11) nodemon server/aution_server.ts