

- [安装Node.js](#)
- [总结](#)

安装Node.js

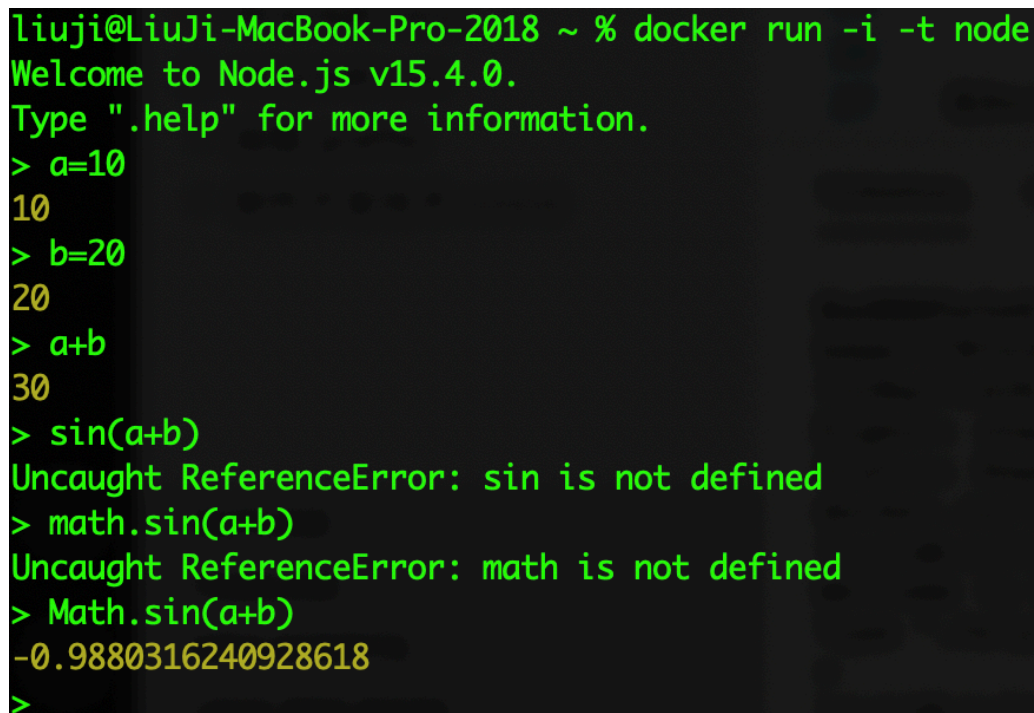
Node.js是一个开源与跨平台的JavaScript运行时环境，它可以让我们在浏览器之外运行JavaScript代码。安装Node.js最简单的方法是使用官方的Docker镜像（几乎不用进行任何操作），如下拉取Node.js：

```
1 | docker pull node
```

然后运行

```
1 | docker run -i -t node
```

进入交互式环境

A terminal window with a dark background and green text. The prompt is 'liuji@LiuJi-MacBook-Pro-2018 ~ %'. The command 'docker run -i -t node' has been executed. The terminal shows the Node.js welcome message 'Welcome to Node.js v15.4.0. Type ".help" for more information.' followed by several commands and their outputs: '> a=10' outputs '10', '> b=20' outputs '20', '> a+b' outputs '30', '> sin(a+b)' outputs an error 'Uncaught ReferenceError: sin is not defined', '> math.sin(a+b)' outputs an error 'Uncaught ReferenceError: math is not defined', and '> Math.sin(a+b)' outputs '-0.9880316240928618'. The prompt '>' is visible at the bottom.

```
liuji@LiuJi-MacBook-Pro-2018 ~ % docker run -i -t node
Welcome to Node.js v15.4.0.
Type ".help" for more information.
> a=10
10
> b=20
20
> a+b
30
> sin(a+b)
Uncaught ReferenceError: sin is not defined
> math.sin(a+b)
Uncaught ReferenceError: math is not defined
> Math.sin(a+b)
-0.9880316240928618
>
```

如果要运行物理机上编写JavaScript代码，就需要用'-v' 指令加载本地目录

另一种方式是在Linux镜像中安装Node.js， [Dockerfile](#)如下：

```
1 FROM centos
2 # 安装Node.js 14
3 RUN dnf module -y install nodejs:14
4 #新建/root/scripts目录
5 RUN mkdir -p /root/scripts
6 #将scripts目录中的代码拷贝到/root/scripts
7 COPY scripts /root/scripts
```

在Dockerfile所在目录执行

```
1 docker build -t centos-nodejs:latest .
```

创建镜像。之后执行

```
1 docker run -i -t centos-nodejs
```

进入镜像。可以在 `/root/scripts` 目录下找到[hello.js](#)这个JavaScript文件，hello.js的内容如下：

```
1 function add(a,b)
2 {
3     return a+b;
4 }
5 var str='hello world';
6 console.log(str);
7 console.log("1+2=",add(1,2));
8
```

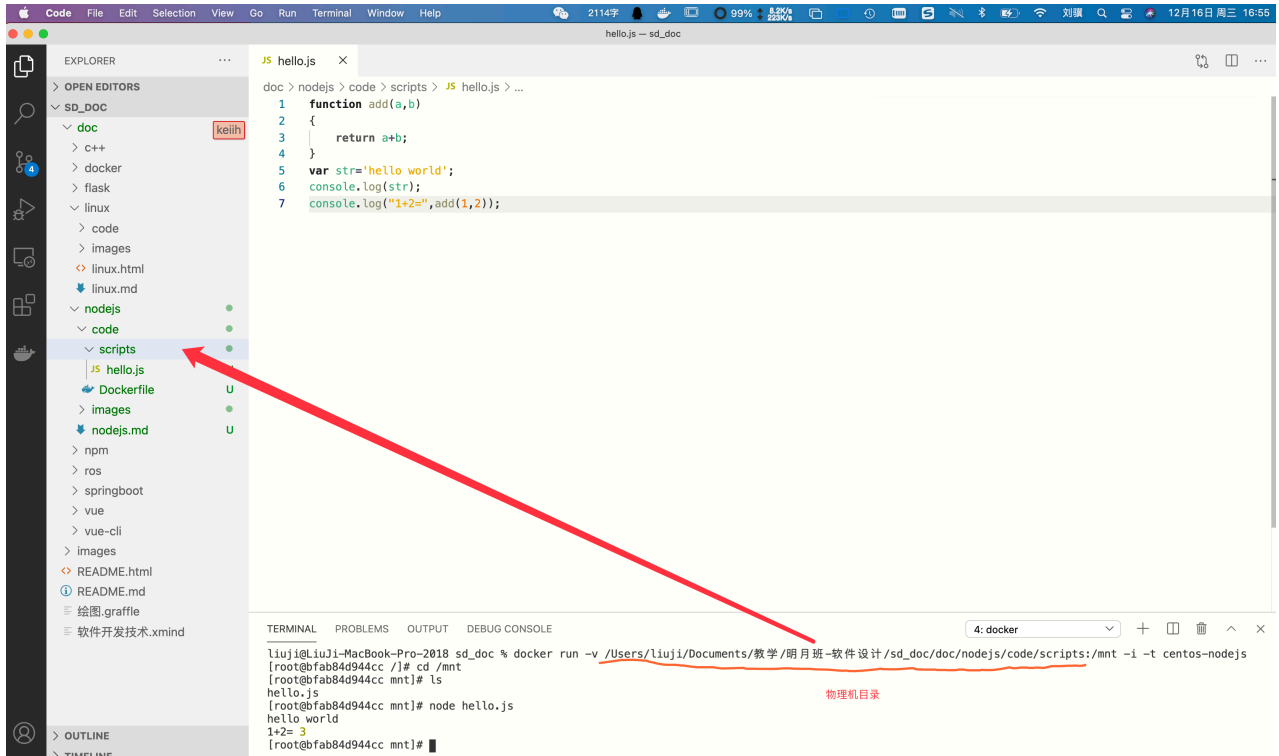
在命令行下执行

```
1 node hello.js
```

即可执行该程序，结果如下：

```
liuji@LiuJi-MacBook-Pro-2018 ~ % docker run -i -t centos-nodejs
[root@ad24ce886545 /]# node /root/scripts/hello.js
hello world
1+2= 3
```

你可以用 `-v` 参数将物理机目录挂载到虚拟机上，然后就可以用物理机的VSCode编辑JavaScript代码，并在虚拟机上运行，如下图所示：



总结

Node.js是NPM工具的基础，要使用NPM必须先安装Node.js。我们并不会直接使用Node.js，而是更多的使用NPM和vue-cli。所以只要你能运行上面的Docker镜像，就算是掌握了。如果想用Node.js进行服务器端编程（通常我们会选择Python或者Java，但如果非要用JavaScript.....）可以完整的学习一下[Node.js的官方文档](#);