



**05项目部署文档**

**《互联网数据库开发》课程团队作业**

**小组成员：姜宇，唐明昊，王禹衡，徐海潆**

**学号：2210705、2113927、2213040、2212180**

# 前端部署：

## Node.js：

### 什么是Node.js:

在之前大二的专业选修课《数据可视化》中，我们已经接触过对于Node.js的配置和使用了。

[Node](https://so.csdn.net/so/search?q=Node&spm=1001.2101.3001.7020" \t "https://blog.csdn.net/thlzjfefe/article/details/_blank).js 是一个基于 Chrome V8 引擎的 JavaScript 运行环境。Node.js 使用了一个事件驱动、非阻塞式 I/O 的模型，使其轻量又高效。简单地说 Node.js 就是运行在服务端的 JavaScript，利用JavaScript在服务端进行编程。Node.js是一个让JavaScript运行在服务器端的开发平台，它让JavaScript的触角伸到了服务器端，可以与PHP、JSP、Python、Ruby平起平坐。Node.js不是一种独立的语言，与PHP、JSP、Python、Perl、Ruby的“既是语言，也是平台”不同，Node.js使用JavaScript进行编程，运行在JavaScript引擎上（V8）。nodejs开发的构成就是利用[npm](https://edu.csdn.net/cloud/sd_summit?utm_source=glcblog&spm=1001.2101.3001.7020" \t "https://blog.csdn.net/thlzjfefe/article/details/_blank)开发的社区提供的大量的第三方包加上基本的ECMAScript脚本语言以及node平台提供的一系列编程接口进行编程。

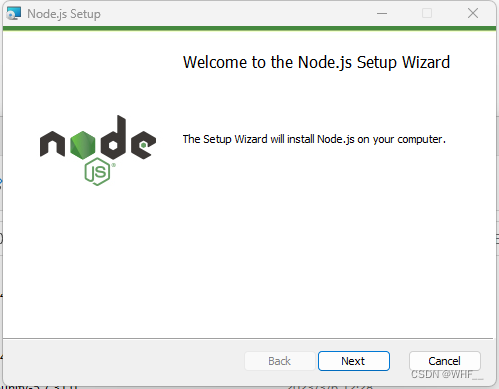
### Node.js安装部署：

#### 进入官方地址网站下载安装包：

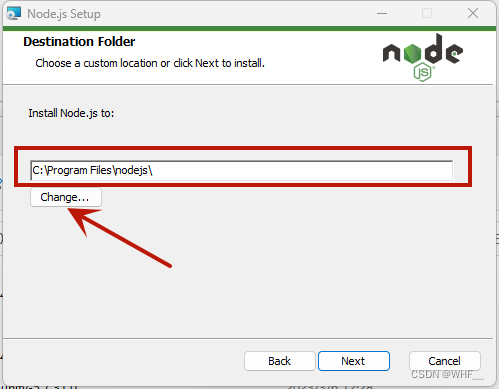


#### 安装程序：

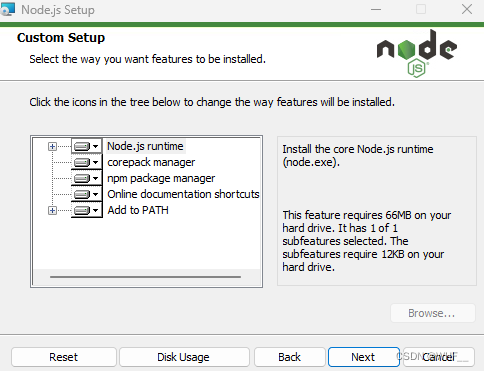
**·首先打开安装包，开始安装Node.js:**



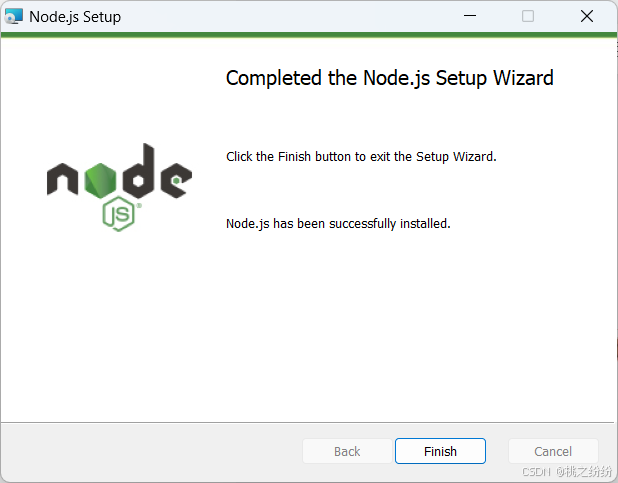
**·选择对应的路径：**



**·而后根据自身需求进行选择：**



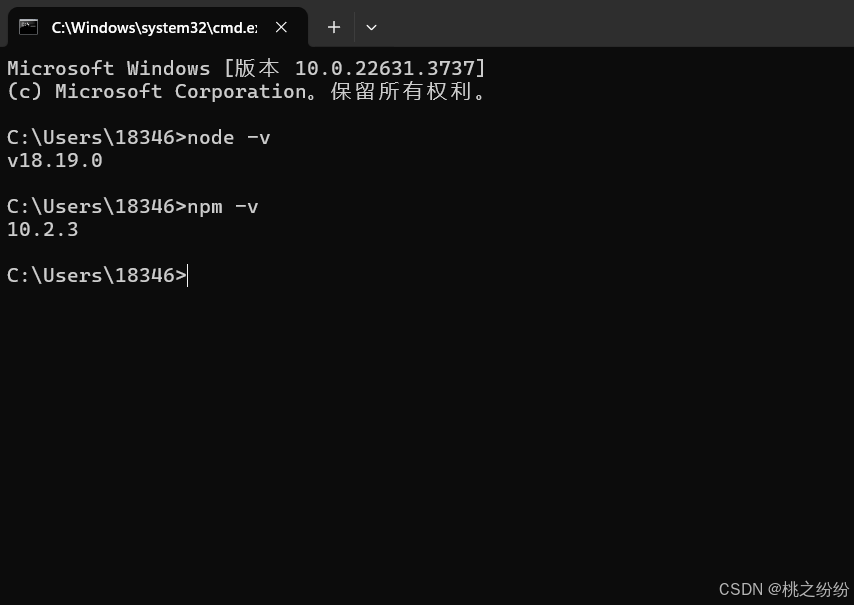
**·点击Finish，完成安装：**



**·检验：测试安装是否成功，按下【win+R】键，输入cmd，打开cmd窗口：**

输入：node -v     // 显示node.js版本

[npm](https://edu.csdn.net/cloud/sd_summit?utm_source=glcblog&spm=1001.2101.3001.7020" \t "https://blog.csdn.net/WHF__/article/details/_blank) -v      // 显示npm版本



## 工具包下载：

在完成了对于node.js的安装之后，我们就可以完成对于前端代码的运行了，但是，在此之前，我们需要先下载，我们网页中所使用到的多种工具包。

与Python类似，在Python中，如果我们想要使用某个工具包，除了需要在代码中import之外，还需要下载对应的安装包，使用pip指令。那同样的，在node.js中，我们使用npm install来下载对应的工具包。具体的代码如下所示。

**npm install**

**npm run format**

**npm install element-plus**

**npm install plyr**

**npm install comment-core-library**

**npm install gsap**

**npm install axios**

**npm install ip**

## 前端启动：

在下载完成所有的安装包之后，我们就可以运行前端代码了。

在终端中，cd到frontend文件夹，而后输入指令：

**npm run dev**

即可实现前端代码的运行。

# 后端部署：

在完成了前端的代码部署之后，我们输入指令开始运行。在进入对应的登录注册页面之后，我们会发现，无法注册或者登录。这是由于我们后端还没有运行，无法提供相应的服务。

下面我们进行后端代码的配置，我们本次作业使用了yii2框架完成对于网络的开发。因此，下面提供了后端代码的配置方法：

## Composer配置：

如果还没有安装 Composer，你可以按 [getcomposer.org](https://getcomposer.org/download/) 中的方法安装。 在 Linux 和 Mac OS X 中可以运行如下命令：

**curl -sS https://getcomposer.org/installer | php**

**mv composer.phar /usr/local/bin/composer**

在 Windows 中，你需要下载并运行 [Composer-Setup.exe](https://getcomposer.org/Composer-Setup.exe" \o "Composer-Setup.exe)。

在本指南中，所有 composer 命令都假定您已经安装了[全局](https://getcomposer.org/doc/00-intro.md" \l "globally" \o "全局) 的 composer， 这样它可以作为 composer 命令。如果您在本地目录中使用 composer.phar， 则必须相应地调整示例命令。

## yii配置：

在完成了对于composer的安装之后，那么接下来就需要完成对于yii框架的需求的安装了。

我们已经完成对于composer的安装。因此类比node.js中对于安装包的下载。

（1）首先cd到backend的文件夹下，而后输入指令**composer install**完成对于各种工具包的安装，需要注意的是：



（1）如果你成功地完成了所有安装包的下载。之后，应该可以运行后端的代码了。我们输入指令：

**php yii serve --port <端口号>**

不出意外的话，应该会输出以下信息：

**Server started on http://localhost:<端口号>**

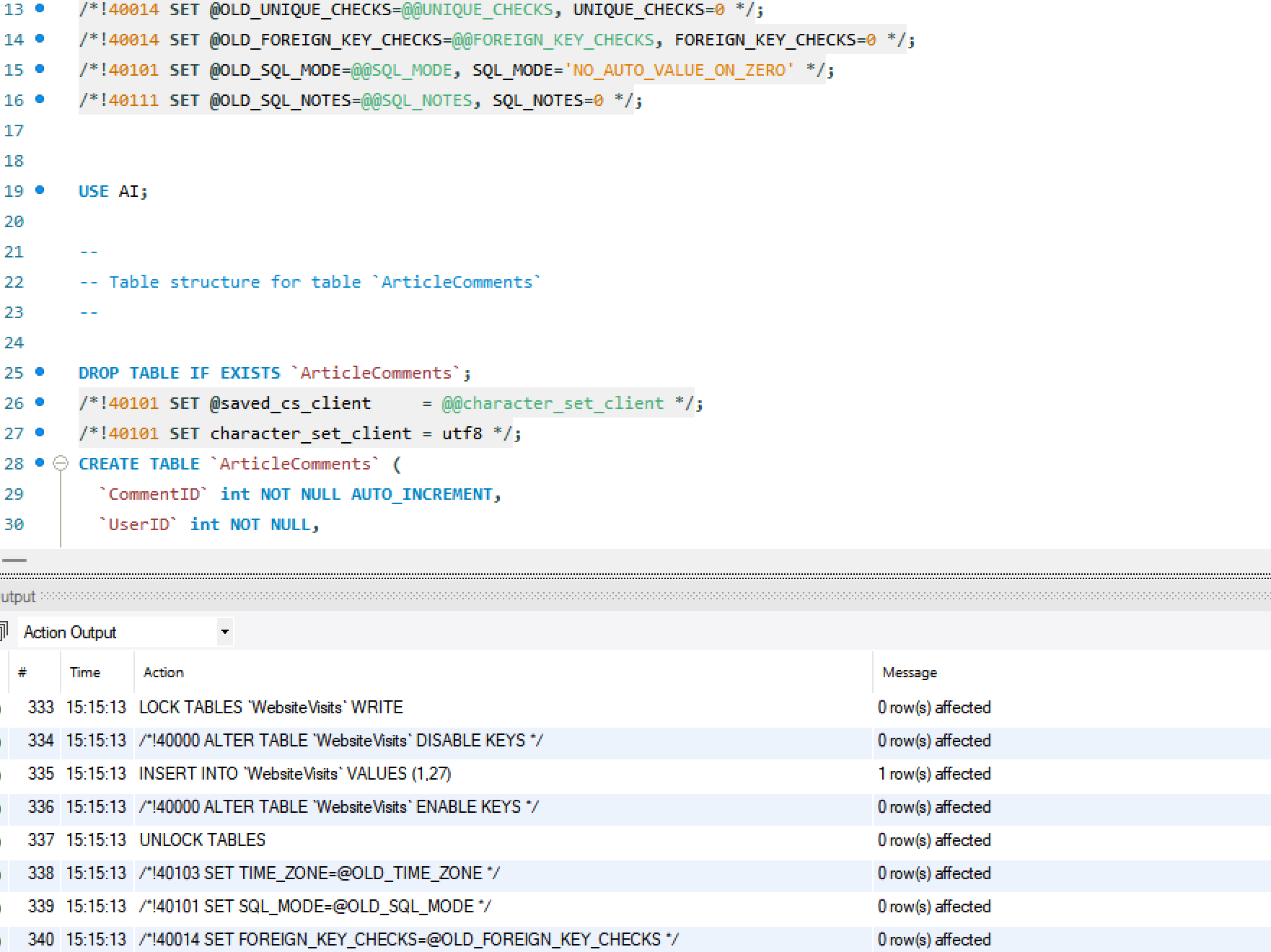
而后访问对应的网站，就可以实现后端管理页面的访问。

# 数据库部署：

考虑到我们在大二下学期无论是在《数据库系统》理论课程的学习中，还是在上机考试中，都已经学习并使用过mysql或者xampp对应的功能和方法。因此，关于mysql的配置，可以参考：

[https://blog.csdn.net/unauna9739/article/details/124702155.](https://blog.csdn.net/unauna9739/article/details/124702155。)

我们在实验中，需要将对应的sql文件引入到数据库中：



# 一键部署脚本：

我们同样配备了整个程序的一键部署脚本，双击start.bat文件，输入对应的数据库密码和root等信息之后，就可以实现一键部署。

# 程序运行：

·每次运行前，确认已经完成了数据库的导入，并正确连接。

·前端：cd到frontend，输入指令：npm run dev

·后端：cd到backend，输入指令：php yii serve --port <端口号>