**华南农业大学信息学院**

设计性、综合性实验

|  |  |  |  |
| --- | --- | --- | --- |
| **实验题目** | 决战Linux | **起止日期** | **2017-2018学年第2学期** |
| **课程名称** | LINUX系统应用与程序设计 | | |
| **学生资料** | 学号 学生姓名  院系 数学与信息学院 专业班级 | | |
| **实验内容** | 在LAMP、LNMP等环境下，任意挑选下列2种或以上动态网页语言，写出一个页面，读取MYSQL实验中的数据表数据。（PHP，JSP，Perl，Python，Ruby，Go，ASP.NET，Node.JS…）注意：不同的动态网页语言，所要求的具体环境有可能不同，需要自行搭建。 | | |
| **评**  **语**  **与**  **成**  **绩** | |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | **项目/分数** | **优** | **良** | **中** | **及格** | **不及格** | | 系统分析设计 | □ | □ | □ | □ | □ | | 程序设计水平 | □ | □ | □ | □ | □ | | 内容完成情况 | □ | □ | □ | □ | □ | | 设计报告撰写质量 | □ | □ | □ | □ | □ | | 课程设计总结情况 | □ | □ | □ | □ | □ |   附加说明：  成绩 (优/良/中/不及格) 指导教师 李舜鹏  年 月 日 | | |

1、实验步骤描述

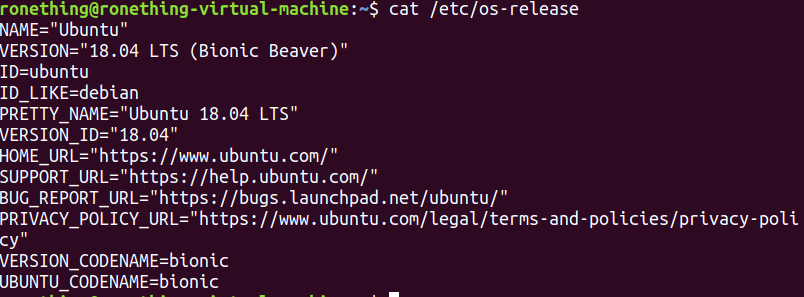
（1）第一种动态语言 python

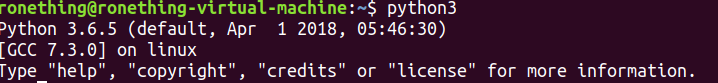
A、实验环境描述

[Ubuntu@18.04](mailto:Ubuntu@18.04) LTS (虚拟机)

[Python@3.6.5](mailto:Python@3.6.5)

[Mysql@5.7.22](mailto:Mysql@5.7.22)



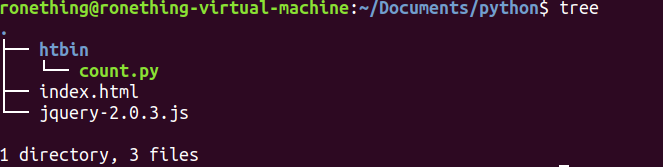




B、系统架构图

C、相关配置总结

目录结构：



Python3-pip python包管理器

PyMySQL PyMySQL 是在 Python3.x 版本中用于连接 MySQL 服务器的一个库

使用python3自带的CGIHTTPServer作为服务器

这里其实要对mysql操作一波， 创建数据库linux，数据库用户linuxuser并且设置密码'17678'，给予其数据库操作权限。

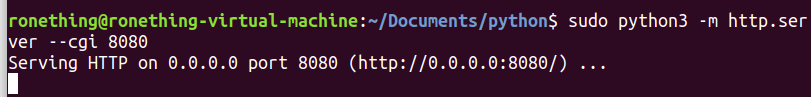
|  |
| --- |
| **sudo mysql 进入mysql控制台**  **Create databases linux;**  **Create user linuxuser;**  **Alter user linuxuser identified by '17678';**  **Grant all privileges on linux.\* to linuxuser identified by "17678";**  **Flush privileges;**  **Quit;** |

D、核心代码总结

|  |
| --- |
| **Index.html**  **<!DOCTYPE html>**    **<head>**  **<title>hello</title>**  **<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />**  **<script src="/jquery-2.0.3.js"></script>**  **<style>**  **table,**  **td {**  **border: 1px solid #111**  **}**  **</style>**  **</head>**    **<body>**  **<button id="hello">查看数据表</button>**  **<script>**  **$("#hello").click(function () {**  **$(this).attr("disabled", "disabled");**  **var html = $.ajax({**  **async: true,**  **url: "/htbin/count.py",**  **dataType: "json",**  **success: function (data) {**  **var table = document.createElement('table'),**  **string = "";**  **for (var key in data) {**  **string = "";**  **string = "<tr><td>" + key + "</td>" +**  **"<td>" + data[key] + "</td></tr>";**    **table.innerHTML += string;**  **}**    **document.body.appendChild(table);**  **console.log(data);**  **},**  **error: function (data) {**  **console.log(data);**  **}**  **})**  **})**    **</script>**  **</body>** |
| **Count.py**  **#!/usr/bin/python3**  **# -\*- coding: UTF-8 -\*-**    **import pymysql**  **import json**    **print('Content-Type: text/html')**  **print() #空行，告诉服务器结束头部**  **# 打开数据库连接**    **db = pymysql.connect("localhost","linuxuser","17678","linux")**    **# 使用cursor()方法获取操作游标**  **cursor = db.cursor()**    **# 使用execute方法执行SQL语句**    **sql = "select \* from demo";**    **try:**  **cursor.execute(sql)**  **results = cursor.fetchall()**  **dic = {}**  **for row in results:**  **dic[row[0]] = row[1]**    **print(json.dumps(dic))**  **except:**  **print("Error:unable to fecth data")**    **# 关闭数据库连接**  **db.close()** |

E、页面效果截图

启动服务器：



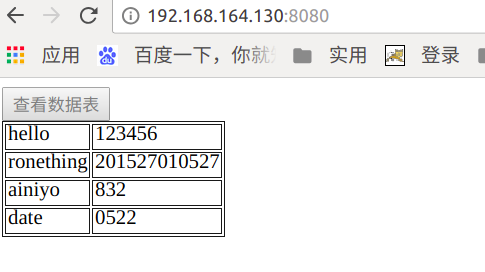
访问前端网页，虚拟机ip为192.168.164.130，所以访问http://192.168.164.130:8080

效果图：

初始界面：



点击按钮后，数据表数据读取出来，按钮变暗（设置为不可点击）。



（2）第二种动态语言 php

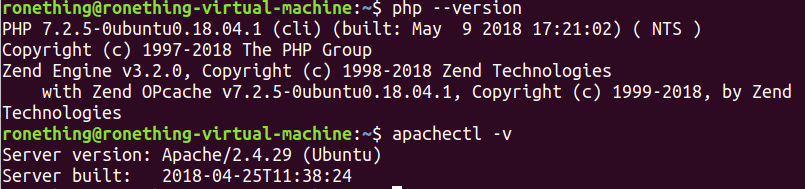
A、实验环境描述

Ubuntu 18.04 LTS(虚拟机)

php@7.2

mysql@5.7.22

[apache@2.4.29](mailto:apache@2.4.29)



B、系统架构图

C、相关配置总结

Lamp环境搭建详情可查看文章 [Ubuntu 18.04 LTS 搭建 LAMP 环境](https://www.cnblogs.com/ronething/p/9176816.html)

作者为本人。

命令如下：

|  |
| --- |
| **sudo apt-get install apache2 –y**  **sudo apt-get install php7.2 -y**  **sudo apt-get install libapache2-mod-php7.2**  **sudo apt-get install mysql-server –y**  **sudo apt-get install php7.2-mysql** |

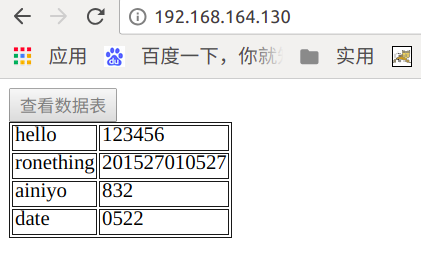
D、核心代码总结

|  |
| --- |
| **index.html 基本和python的index.html一样 修改请求url即可**  **<!DOCTYPE html>**    **<head>**  **<title>hello</title>**  **<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />**  **<script src="/jquery-2.0.3.js"></script>**  **<style>**  **table,**  **td {**  **border: 1px solid #111**  **}**  **</style>**  **</head>**    **<body>**  **<button id="hello">查看数据表</button>**  **<script>**  **$("#hello").click(function () {**  **$(this).attr("disabled", "disabled");**  **var html = $.ajax({**  **async: true,**  **url: "/htbin/linux.php",**  **dataType: "json",**  **success: function (data) {**  **var table = document.createElement('table'),**  **string = "";**  **for (var key in data) {**  **string = "";**  **string = "<tr><td>" + key + "</td>" +**  **"<td>" + data[key] + "</td></tr>";**    **table.innerHTML += string;**  **}**    **document.body.appendChild(table);**  **console.log(data);**  **},**  **error: function (data) {**  **console.log(data);**  **}**  **})**  **})**    **</script>**  **</body>**    **</html>** |
| **Linux.php**  **<?php**    **$con = mysqli\_connect('localhost', 'linuxuser', '17678','linux');**  **if (!$con)**  **{**  **die('Could not connect: ' . mysqli\_connect\_error());**  **}**    **//echo "连接成功\n";**    **$sql="SELECT \* FROM demo";**    **$result = mysqli\_query($con,$sql);**    **$dic = array();**    **while($row = mysqli\_fetch\_assoc($result))**  **{**  **$dic[$row['name']]=$row['id'];**  **}**    **echo json\_encode($dic);**  **mysqli\_close($con);**  **?>** |

E、页面效果截图

Apache 默认开启端口80 所以直接访问 <http://192.168.164.130>





（3）第三种动态语言 node.js

A、实验环境描述

Ubuntu 18.04 LTS(虚拟机)

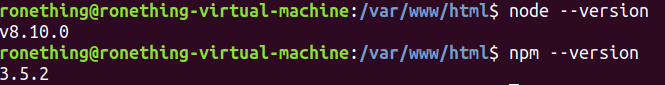
[php@7.2](mailto:php@7.2)

[mysql@5.7.22](mailto:mysql@5.7.22)

[apache@2.4.29](mailto:apache@2.4.29)

[Node@v8.10.0](mailto:Node@v8.10.0)

[Npm@3.5.2](mailto:Npm@3.5.2)



B、系统架构图

C、相关配置总结

安装mysql nodejs 驱动：

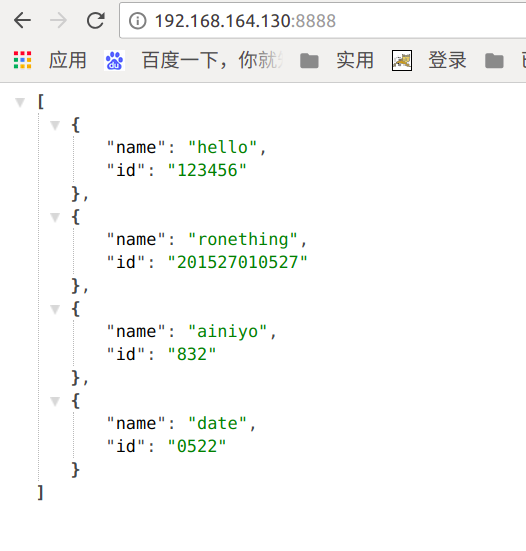
|  |
| --- |
| **npm install mysql** |

D、核心代码总结

|  |
| --- |
| **Index.html**  **<!DOCTYPE html>**    **<head>**  **<title>hello</title>**  **<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />**  **<script src="/jquery-2.0.3.js"></script>**  **<style>**  **table,**  **td {**  **border: 1px solid #111**  **}**  **</style>**  **</head>**    **<body>**  **<button id="hello">查看数据表</button>**  **<script>**  **$("#hello").click(function () {**  **$(this).attr("disabled", "disabled");**  **var html = $.ajax({**  **async: true,**  **url: "<http://192.168.164.130:8888>",**  **dataType: "json",**  **success: function (data) {**  **var table = document.createElement('table'),**  **string = "";**  **for (var key in data) {**  **string = "";**  **string = "<tr><td>" + data[key]['name'] + "</td>" +**  **"<td>" + data[key]['id'] + "</td></tr>";**    **table.innerHTML += string;**  **}**    **document.body.appendChild(table);**  **console.log(data);**  **},**  **error: function (data) {**  **console.log(data);**  **}**  **})**  **})**    **</script>**  **</body>**    **</html>** |
| **Nodejs.js**  **var mysql=require('mysql');**  **var http = require('http');**  **var server=http.createServer(function(res,res){**    **res.writeHead(200,{"Content-Type":"text/plain","Access-Control-Allow-Origin":"http://192.168.164.130"});**    **var connection=mysql.createConnection({**  **host:'localhost',**  **user:'linuxuser',**  **password:'17678',**  **database:'linux'**  **})**  **connection.connect();**  **connection.query('SELECT name,id from demo',function(error,results,field){**  **if(error) throw error;**  **res.write(JSON.stringify(results));**  **res.end();**  **});**    **});**  **server.listen('8888','192.168.164.130',function(){**  **console.log("start listening...");**  **})** |

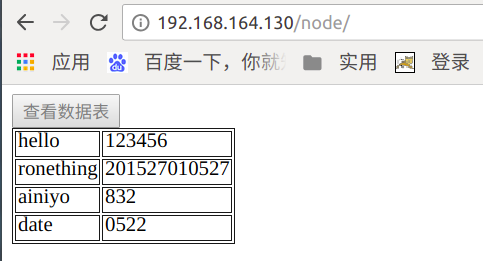
E、页面效果截图

先看8888端口能不能获取数据库数据



没有问题，访问前端页面 <http://192.168.164.130/node/>





2、实验总结

注：文档使用word online 编写。所以导出的doc文档也许会有点难看。。