

马哥教育Python全栈web开发班课程大纲

专业定位	python全栈web开发班旨在指导零基础学员学完该课程后，可熟练应用企业应用开发，拥有中级Python编程开发能力					
课程周期	28天					
上课形式	周六全天在线直播（9:30-12:00 14:00-18:30）					
就业方向	Python全栈开发工程师					
课程阶段	课程时间	课程名称	课程内容简介	核心技术点简介	项目项目	教学目标
	1	Python基础入门	开课 简易linux教程 Python入门，包括变量，逻辑控制，循环	<ul style="list-style-type: none"> * Python 发展历史和语言特性 * Python 环境搭建及虚拟环境管理 * 第一个 Python 程序 * Python 四则运算 * Python 数据结构 * Python 元运算符（in、not in、is、and、or） * 用户输入（input、raw_input） * 流程控制 <ul style="list-style-type: none"> * 缩进 * if..else 条件语句 * for、while 循环语句 * break continue 中断语句 	项目1：统计最大的两个值 项目2：统计字符出现次数	

一、Python 基础语法	2	内置数据结构 (一)	列表特点 for循环 列表操作方法 元组特点	<ul style="list-style-type: none"> * 列表的定义 * 列表的遍历 * 列表的切片 * 常用内置函数 (max、min、range、len) * list 方法 (append、pop、count ...) * 列表表达式 * 元组的定义 * tuple 方法 (count、index) * 元组与列表的关系与区别 * 字符串处理 (find、join、split、strip、format、replace、index、count ...) 	<ul style="list-style-type: none"> * 大战面试官 3: 实现简单的队列和栈 * 大战面试官 4: 冒泡排序讲解 * 项目5: 基于list存储的交互式用户登录验证 	1. 熟练搭建Python开发环境及环境变量配置 2. 熟练掌握Python基础语法 3. 了解Python
	3	内置数据结构 (二)	字典介绍 字典特点 字典和列表比较 集合特点 集合操作方法	<ul style="list-style-type: none"> * 字典的定义和特点 * 字典的遍历 * dict 方法 (get、iteritems、update、keys、values ...) * 大战面试官: 列表的字典的区别和关系, 分别适用于什么场景 * 集合与集合的基本操作 	项目6: IP 维度统计 Nginx 日志 项目7: 工作中遇到日志分析问题, 怎么全方位多维度的来统计日志	
	4	Python文件处理	处理文件方法 文件指针 基于文件存储的简单数据类型	<ul style="list-style-type: none"> * 文件IO操作 * 打开文件 * 文件操作 (write、read、readlines、writelines、seek、flush、close ...) 	* 项目8: 基于文件存储的交互式用户登录注册系统	

5	函数及函数式编程	定义函数 参数 返回值 函数式编程	<ul style="list-style-type: none"> * 函数在干什么的 * 函数的定义 * 函数的返回值 * 位置参数和关键字参数 * 全局变量和局部变量 * 深入函数定义（可变参数列表、参数列表的分拆、Lambda 形式） * 理解函数式编程 * Lambda 匿名函数 * Sorted 排序函数 * 递归 * 匿名函数 * 生成器 	* 项目9:用函数优化之前的代码
---	----------	----------------------------	---	------------------

二、Python 基础进阶

6	类及面向对象编程	<p>面向对象</p> <p>类的定义</p> <p>类的继承</p> <p>错误处理</p> <p>自定义异常类</p>	<ul style="list-style-type: none"> * 类的基本概念与详解 * 变量与方法 * 作用域与命名空间 * 类的继承 * 类的抽象、封装、方法与实例化实践 * 常用内置模块（<code>sys</code>、<code>os</code>、<code>commands</code>） * 脚本参数处理 <code>sys.argv</code> * 时间的控制 <code>time</code> * 邮件模块 <code>smtplib</code> * 封装 * 继承 * 多继承与MRO * <code>Mixin</code> * 运算符重载 * 内置方法与对象的特殊方法 * 描述器 * 对象的生成销毁与元类 	<p>* 项目10：简单游戏实现 *</p> <p>项目11：爬虫爬取知乎信息，给自己发邮件</p>
---	----------	---	---	--

1 熟练使用
Python 2 掌握面向对象思想

	7	Python进阶 模块	多线程多进程 装饰器 目录的遍历和操作	<ul style="list-style-type: none"> * 并发的基本概念 * 多线程编程基础 * 锁、事件和信号量 * 线程池 * 多进程与多线程的异同 * 装饰器 * 带参数的装饰器 * 装饰器的应用 * 文件的打开和关闭 * 读写文件 * context与with语法 * BytesIO、StringIO和类文件对象 * 文件和目录相关操作 	* 项目12：实现多线程下载器	
	8	模块化编程及Django入门	基于文件存储实现一个留言板 settings配置 url配置 view逻辑	Django作为开源Web框架，和Flask并驾齐驱颇为出名，是重量级架构的代表： <ul style="list-style-type: none"> * setting配置 * url配置 * ORM * filter * cookie * 模板渲染 	* 项目13：基于文件存储，实现Web端备忘录	

三、Python 高级用法	9	Python数据库操作	数据库概述 SQL语句学习 Python链接MySQL 实现留言板的增删改查	* 流行数据库介绍 * SQL 语句学习 * 数据库的创建、删除 * 表结构创建、删除、修改 * 数据的增添改查(insert、delete、update、select) * Python 操作 MySQL * 大战面试官：高性能数据库系统漫谈 * 项目1：用 Flask 实现数据库的增添改查 * 基础功能（可以简单的实现Web增添改查功能） * 项目目标（巩固Flask、MySQLdb基础	* 项目13进阶：基于 MySQL 存储的 WEB 端备忘录	熟练使用Django 开发常见Web系统
	10	Django进阶	form验证 session+cookie登陆验证 admin	* Django高级使用 * Django admin使用 * 登陆认证 * Django体系了解	* 项目13进阶：备忘录实现权限认证	

	11	前端基础 (一)	html入门 css入门 bootstrap框架使用	<ul style="list-style-type: none"> * HTML 学习 (table、表单、列表 ... 等) * CSS 简介 (盒模型) * Bootstrap 框架学习 <ul style="list-style-type: none"> * 栅格系统 * 按钮组 * 表格样式 * 表单样式 * 前端学习JQuery <ul style="list-style-type: none"> * 查找一个元素 (选择器) * 操作元素内容 (html) * 修改元素属性 (attr、css) * 操作输入框 (val) * 练习: 聊天窗效果模拟 * 项目1: 基于 MySQL 存储的 WEB 用户登录注册系统 <ul style="list-style-type: none"> * 支持增删查 * 要求使用Bootstrap样式呈现登录和 	* 项目13进阶: 使用bootstrap 美化备忘录	
--	----	-------------	--	---	-----------------------------------	--

四、Python Web开发	12	前端基础 (二)	js入门 jquery入门 jquery选择器 jquery操作函数 jquery动画	* JQuery 简介和应用 * JQuery 核心思想 * JQuery 核心模块介绍及使用 * Ajax和Web2.0介绍是什么 * Json 是什么？Json 与字典有哪些相似？ * JQuery 与 Flask 异步请求交互 * 常见JQuery 插件，实现前端展示效果 * 模态弹窗 * 消息弹窗 SweetAlert * JQuery UI 丰富的组件，日期选择、小滑块等 * Datatables 表格个性化展示分页、搜索等 * Flask 登录模块 * 登录功能的完善 * 项目1：基于 MySQL 存储的 WEB 用户登录注册系统（支持增删查，要求使用Bootstrap 样式展现登录和列表） * 支持增删查 * 要求使用Bootstrap 样式展现登录和列表 * 结合当天学的Ajax、JQuery知识点，把页面请求改成异步	* 项目14进阶：使用ajax实现备忘录的异步操作	掌握前端基础，前后端打通，成为全栈开发
	13	前端基础 (三)	jquery插件 bootstrap的js插件使用 sweetalert	完善系统的细节，报错错误提示，表单验证和弹窗，实现备忘录的动态效果		

	14	【项目1】 大型电子商务购物商城（类天猫商城）	需求分析 用户管理，商户段，商品列表 设计表结构 设计Django app	* 用户所需字段分析 * 商户字段分析 * 商品字段分析 * 表结构设计注意事项 * 根据需求设计不同的Django app	* 项目15：大型类京东电商项目实战	
	15	【项目2】 商务网站会员管理系统（类京东）	登陆 注册 用户管理	* 登陆需求分析 * 密码加密存储 * 加密算法简介 * 密码重置方法		
	16	【项目3】 B2B商品电商商家后台管理系统（类阿	商品的管理 Django模板继承	* 多页面继承关系 * 商品的增删改查		
	17	商品页列表展示（类唯品会）	商品列表展示页 商品详情页 访问次数统计	* 列表展示页和详情页 * 访问次数统计实现（实战的几个详情可以合并在一起）		

五、Python
大型案例应用
进阶

18	【项目4】 Python高级 爬虫（应 用于知乎 、豆瓣）	http库requests 解析html的 pyquery 使用爬虫批量入库 商品	* urllib * urllib2 * requests * htmlparser * threading * gevent * http https ftp协议 * IP限制 * 验证码原理	* 项目16：爬虫 项目	
19	【项目5】 年度账单 系统之商 户图表统 计（类支 付宝）	Echarts入门 柱状图 折线图 饼图 其他使用	* 可视化介绍 * Echarts介绍 * 图表所需模块使用	* 项目17：使 用Echarts完成 常见图表的实 现（假数据）	1. 掌握常见网络 协议 2. 掌握常见 Python网络库和 解析库 3. 掌握高并发， 多线程程序 4. 熟悉常见的反 爬虫机制及原理 和处理方法 5. 完成批量商品 入库项目
20	【项目6】 网站数据 可视化系 统（类 CNZZ、百 度统计）	Echarts使用 实现可视化展示浏 览量	* 动态数据渲染Echarts * 实现所需报表和图表 * Python到处excel的实现	* 项目17进 阶：使用数据 库数据渲染 Echarts图表	

21	大型银行 网站代码 架构调优	细节优化 搜索功能 全局404和500页面 常见SQL注入，XSS 攻击防御	* 一个Web系统所需的完整组件 * 常见额外功能的实现方式 * 常见安全漏洞和防护措施 * 如何完成项目的部署	* 项目17进阶：完善整个实战系统，可以上线使用
22	【项目7】 点餐排队 系统技术 之Redis (类饿了)	Redis介绍 Redis数据类型 Redis使用	* Redis应用场景 * Redis数据类型	
23	消息队列 与celery	理解Celery 指定队列 任务绑定 记录日志	* 为什么需要队列 * Celery如何使用 * Celery特性	
24	MySQL进阶 实战	数据库设计范式 SQL性能分析 高可用数据库策略	* 数据库基本操作 * 自动生成SQL * 结果集映射为对象 * SQLAlchemy 基本操作	

	25	【项目8】 大型招聘网站（类拉勾网）	前端发展潮流 mvvm设计理念 vuejs入门 vuejs组件化 react简介 了解前	<ul style="list-style-type: none"> * 前端技术体系 * vue核心概念 * component * filter * 数据绑定 * vue技术周边 * 单页面应用 	* 项目18：仿拉勾网做一个单页应用	
六、Python架构师之路	26	IT进阶必备之基础	算法：排序，查找，哈希，图算法简介 网络协议：ip tcp http	<ul style="list-style-type: none"> * 冒泡排序 * 快速排序 * dfs * bfs * 网络协议关系 * socket 		如何成为一个高级程序员必经之路
	27	IT进阶之自我修养	代码规范 Web开发项目流程：敏捷 代码评审	<ul style="list-style-type: none"> * 什么是好代码 * 常见Python代码规范 * 代码规范检查工具 * 常见项目开发流程 * 项目可维护性 		
	28	面试必备	什么是好的简历 常见面试题 面试软技能	<ul style="list-style-type: none"> * 好的简历的特性 * 突出重点 * 常见面试题 * 如何自我介绍 * 如何谈薪资 * 如何在技术之外吸引面试官 		