**《Python程序设计》教学大纲**

**一、课程概况**

|  |  |  |  |
| --- | --- | --- | --- |
| 课程名称（中文） | Python程序设计 | 课程代码 | 050522073 |
| 课程名称（英文） | Programming in Python | 课程属性 | 专业选修课 |
| 学 时 | 48 | 学分 | 2.5 |
| 开课单位 | 金融与数学学院 | 开课学期 | 5 |
| 适用专业 | 金融工程 | 是否核心课 | 否 |

**二、课程描述**

Python目前是世界最流行的编程语言之一，其语法简单、但功能强大，编写简洁且可读性好。Python的另一重要优势是能够与各种编程语言对接，因此它能够用简单的语法结构封装其他编程语言最优先的程序代码。Python目前广泛的应用到金融工程、金融科技、互联网金融等领域。本门课程是一门基础课，适用于没有接触过任何编程语言的学生。通过本门课的学习，让学生了解程序设计的基本方法，掌握Python语言基本语法，能够应用到常见的金融模型当中。

**三、课程目标**

| **课程目标** | **目标要求** | **权重** |
| --- | --- | --- |
| 课程目标1 | 系统掌握Python程序设计的原理及过程，熟悉Python运行原理，理解利用程序设计语言解决实际问题的过程和思路；能够运用Python进行简单的程序设计并应用到常见的金融模型分析中；能够运用Python进行简单的数据处理。 | 0.6 |
| 课程目标2 | 通过Python的一些程序设计培养学生“计算思维”，让学生能够深刻理解问题的 特性并善于利用计算机解决问题；将Python与金融工程原理相结合，让学生尝试解决包括金融产品定价、交易策略设计、金融风险管理等实际金融问题。 | 0.4 |

**四、课程目标与毕业要求指标点对应关系**

|  |  |  |
| --- | --- | --- |
| **课程目标** | **支撑的毕业要求** | **支撑的毕业要求指标点** |
| 课程目标1 | 4.知识运用（M） | 能熟练使用各种金融行业软件与数据分析软件，熟悉基本的金融模型，并可以熟练编写应用程序对金融数据进行分析，熟练掌握数据库工具的使用，具有较强的金融定量分析能力。 |
| 5.技术融合（H） | 技术融合。熟悉office的高级应用，掌握宏编程技术，掌握Python等高级程序语言设计方法，熟练应用现代信息技术，掌握数据分析方法。 |
| 课程目标2 | 2.专业知识（L） | 具有扎实的自然科学基础知识和经济学基础知识，掌握经济学、金融学和金融工程的基础理论与基本知识，具备扎实的数学、统计学与计算机基础，熟悉金融方面的有关方针、政策和法律法规，了解金融工程的发展方向和最新成就。 |
| 4.知识运用（M） | 能熟练使用各种金融行业软件与数据分析软件，熟悉基本的金融模型，并可以熟练编写应用程序对金融数据进行分析，熟练掌握数据库工具的使用，具有较强的金融定量分析能力。 |
| 5.技术融合（H） | 技术融合。熟悉office的高级应用，掌握宏编程技术，掌握Python等高级程序语言设计方法，熟练应用现代信息技术，掌握数据分析方法。 |

**五、课程教学内容**

|  |  |
| --- | --- |
| **第一章 概论** | |
| 课程目标支撑关系 | 课程目标1、2 |
| 教学目标 | 掌握Python相关的背景知识 |
| 教学重点 | 了解Python的背景知识及相关特点；了解Python在金融大数据中的应用 |
| 教学难点 | 让学生接受新的编程软件的思想 |
| 学 时 | 2学时。 |
| 教学方法 | 讲授法、举例法 |
| 主要内容 | 1.计算机的概念  2.计算机编程  3. Python的概述  4. Python在金融大数据中的应用  5. 第一个程序“Hello World” |
| 学习方法 | 自主学习 |
| **第二章 Python变量和基本数据类型** | |
| 课程目标支撑关系 | 课程目标1、2 |
| 教学目标 | 掌握Python变量和基本数据类型；理解数字类型的操作 |
| 教学重点 | 变量和基本数据类型 |
| 教学难点 | 数字类型的操作 |
| 学 时 | 6学时。 |
| 教学方法 | 讲授法、案例法 |
| 主要内容 | 1. 变量的命名和使用  2. 数值  3. 字符串  4. 元组  5. 集合  6. 字典  7. Python相关的公有方法 |
| 学习方法 | 自主学习 |
| **第三章 Python的列表** | |
| 课程目标支撑关系 | 课程目标1、2 |
| 教学目标 | 掌握列表是什么以及如何使用列表元素；掌握元组方法 |
| 教学重点 | 掌握如何使用列表元素与元组；理解列表与元组的区别 |
| 教学难点 | 理解代表性启发理论。 |
| 学 时 | 4学时。 |
| 教学方法 | 讲授法、案例法、举例法 |
| 主要内容 | 1. 列表是什么  2. 修改、添加和删除元素  3. 组织列表  4. 使用列表时避免索引错误  5. 操作列表 |
| 学习方法 | 自主学习 |
| **第四章 IF判断语句** | |
| 课程目标支撑关系 | 课程目标1、2 |
| 教学目标 | 掌握条件测试，以检查感兴趣的任何条件。学生将学习简单的if语句，以及创建一系列复杂的if语句来确定当前到底处于什么情形。 |
| 教学重点 | 掌握if语句的编写 |
| 教学难点 | 掌握多种类型的if语句 |
| 学 时 | 4学时。 |
| 教学方法 | 讲授法、案例法 |
| 主要内容 | 1. 一个简单示例  2. 条件测试  3. if语句  4. 使用if语句处理列表  5. 设置if语句的格式 |
| 学习方法 | 自主学习 |
| **第五章 For循环与While循环** | |
| 课程目标支撑关系 | 课程目标1、2 |
| 教学目标 | 掌握for循环中执行更多的操作；掌握如何接受用户输入，让程序能够对其进行处理；如何让程序不断地运行，让用户能够根据需要出入信息，并在程序中使用这些信息 |
| 教学重点 | For循环语句的使用；while循环语句的使用 |
| 教学难点 | 如何编写for和while语句 |
| 学 时 | 4学时。 |
| 教学方法 | 讲授法、案例法、举例法。 |
| 主要内容 | 1. 在for 循环中执行更多的操作 2. 函数input（）的工作原理 3. While循环简介 4. 使用while循环来列表和字典 |
| 学习方法 | 自主学习 |
| **第六章 函数** | |
| 课程目标支撑关系 | 课程目标1、2 |
| 教学目标 | 掌握编写函数；掌握向函数传递信息的方式，学习如何编写主要任务是显示信息的函数，还有用于处理数据并返回一个或一组值的函数。 |
| 教学重点 | 掌握函数的编写方法 |
| 教学难点 | 如何编写函数 |
| 学 时 | 4学时。 |
| 教学方法 | 讲授法、案例法 |
| 主要内容 | 1. 函数定义 2. 传递实参 3. 返回值 4. 传递列表 5. 传递任意数量的实参 6. 将函数存储在模块中 7. 函数编写指南 |
| 学习方法 | 自主学习 |
| **第七章 科学计算包Numpy** | |
| 课程目标支撑关系 | 课程目标1、2 |
| 教学目标 | 导入并使用Numpy创建数组；数组的运算、切片、连接及存取、排序与搜索；数组相关属性与方法；矩阵及线性代数运算。 |
| 教学重点 | 数组的切片、连接、改变形态。数组的相关方法 |
| 教学难点 | 数组的切片及改变形态、线性代数运算 |
| 学 时 | 2学时。 |
| 教学方法 | 讲授法、案例法 |
| 主要内容 | 1. 了解Numpy及导入使用； 2. 理解数组的创建、切片、连接、存取、排序及搜索相关技能； 3. 掌握数组灵活切片的方法及数组连接、排序、搜索相关知识。 |
| 学习方法 | 自主学习 |
| **第八章数据处理包Pandas** | |
| 课程目标支撑关系 | 课程目标1、2 |
| 教学目标 | 导入并使用Pandas创建数据框和序列；数据框和序列相关属性、方法的介绍及使用；数据框和序列的访问、切片及运算；外部数据文件的读取及滚动计算函数的使用 |
| 教学重点 | 数据框、序列相关属性、方法的应用，数据框和序列数据的访问、切片及相互之间的转换。常用外部数据文件的读取。 |
| 教学难点 | 数据框、序列的访问及切片。数据框、序列、数组、列表相关数据结构之间的相互转换。 |
| 学 时 | 2 学时。 |
| 教学方法 | 讲授法、案例法 |
| 主要内容 | 1. 了解Pandas导入及创建数据框和序列 2. 理解数据框和序列的访问、切片及方法 3. 掌握数据框和序列相关方法的灵活应用，数据框、序列、数组、列表之间的相关转换及运用。 |
| 学习方法 | 自主学习 |

|  |  |
| --- | --- |
| **第九章 金融计算器的编写** | |
| 课程目标支撑关系 | 课程目标1、2 |
| 教学目标 | 掌握编写一些简单的Python函数来完成常见的金融计算器，例如货币的时间价值等问题。 |
| 教学重点 | 编写程序解决简单的金融计算问题 |
| 教学难点 | 将简单的金融模型与程序编写相结合 |
| 学 时 | 4 学时。 |
| 教学方法 | 讲授法、案例法 |
| 主要内容 | 1. 编写不需要保存的Python函数 2. 函数的输入参数及预设值 3. 在Python编辑器里定义函数 4. 计算年金 5. 利率换算 6. 连续复利利率 7. 净现值和净现值法则 8. 投资回收期和投资回收期法则 9. 内部收益率 |
| 学习方法 | 自主学习 |

**六、教学要求**

1. 教学中重点使用实例教学法与讨论形式，让学生对抽象的代码理解更加透彻，通过一系列简单的编程示例让学生对基础语法理解更到位。通过讨论，教师能最大限度地了解和掌握学生个体和总体的知识准备程度和认识状况，随时调节教学进程，加强教学的针对性和有效性。学生能在讨论中听取别人的发言并作比较，取长补短，扩大视野，有利于新型师生关系和同学关系的建立。

2. 课堂讲授实行启发式，力求少而精，突出重点与难点。重视对学生的学习方法指导和课堂教学效果信息的反馈，同时将结合课程目标要求，做好考核内容设计，并严格按照本大纲要求做好出勤率统计、作业评价等各项工作。

3. 坚持课后练习是教好、学好本课程的关键。根据正常教学进度布置一定量的课后作业，要求学生按时完成。

**七、实验项目设置及学时分配**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **项目编号** | **实验项目名称** | **实验**  **类型** | **开出**  **要求** | **学时**  **分配** | **每组**  **人数** |
| 0505220731 | Python的基本操作 | 验证 | 必修 | 2 |  |
| 0505220732 | Python变量和基本数据操作 | 验证 | 必修 | 4 |  |
| 0505220733 | Python的列表与元祖编写 | 验证 | 必修 | 2 |  |
| 0505220734 | 判断语句与循环语句的编写 | 验证 | 必修 | 2 |  |
| 0505220735 | 简单的金融模型编写 | 综合 | 必修 | 4 |  |
| 0505220736 | 计算看涨期权价格的程序编写 | 综合 | 必修 | 2 |  |
| **学时总计** |  |  |  | 16 |  |

**八、考核方式与成绩评定**

（一）成绩构成

1.

2．平时成绩说明

平时成绩为百分制，由实验报告（a1）、平时出勤成绩（a2）、课堂表现成绩构成（a3）。

（2）平时成绩评分细则

| **考核项** | **考核说明** | **0-59**  **（不及格）** | **60-69 （及格）** | **70-79**  **（中）** | **80-89**  **（良）** | **90-100**  **（优）** |
| --- | --- | --- | --- | --- | --- | --- |
| 平时作业a1 | 考核学生宏观经济学知识点掌握情况，纸质作业，作业档案上传至网络教学平台，以a1×40%进行平时成绩核算。 | 历次作业平均分≤59分为不及格。 | 历次作业平均分60-69分为及格。 | 历次作业平均分70-79分为中。 | 历次作业平均分80 -89为良。 | 历次作业平均分≥90分为优秀。 |
| 课堂考勤、课堂纪律a2 | 严格考勤制度和课堂纪律，每次课前使用网络教学工具进行考勤，满勤100分，每旷课一次扣10分，违反课堂纪律一次扣5分，情节严重扣10分，以a2×10%进行平时成绩核算。 | 旷课、严重违反课堂纪律4次以上该项0分。 | 旷课、严重违反课堂纪律4次该项60分。 | 旷课、严重违反课堂纪律3次该项70分。 | 旷课、严重违反课堂纪律2次记为80分。 | 满勤100分，每旷课1次记为90分。 |
| 课堂表现a3 | 基础分设置60分，通过课堂回答问题、课后线上交流的点名提问、抢答等环节进行，根据题目难易程度、互动表现核算加、减分，以实际得分a3×10%进行平时成绩核算。 | 得分≤59分为不及格。 | 得分60-69为及格。 | 得分70-79为中。 | 得分80-89为良。 | 得分≥90分为优秀。 |

（3）平时成绩与课程目标的对应关系

| **目标编号** | **课程目标内容** | **平时作业** | **课堂考勤** | **课堂表现** | **分值小计** | **分值比例** |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 课程目标1:系统掌握Python程序设计的原理及过程，熟悉Python运行原理，理解利用程序设计语言解决实际问题的过程和思路；能够运用Python进行简单的程序设计并应用到常见的金融模型分析中；能够运用Python进行简单的数据处理。 | 25 | 30 | 15 | 70 | 70% |
| 2 | 课程目标2：通过Python的一些程序设计培养学生“计算思维”，让学生能够深刻理解问题的 特性并善于利用计算机解决问题；将Python与金融工程原理相结合，让学生尝试解决包括金融产品定价、交易策略设计、金融风险管理等实际金融问题。 | 15 | 10 | 5 | 30 | 30% |
| 分值合计 |  | 40 | 40 | 20 | 100 |  |
| 分值比例 |  | 40% | 40% | 20% |  | 100% |
|  |  |  |  |  |  |  |

（二）课程达成度分析

课程目标达成度评价包括课程分目标达成度评价和课程总目标达成度评价。

1. 课程分目标达成度计算

式中：Di——编号为i的课程目标的达成度；

Sk——支撑编号为i的课程目标有k个考核环节，Sk为编号为k的考核环节的平均得分；

Tk----支撑编号为i的课程目标有k个考核环节，Tk为编号为k的考核环节的总分；

Wk——编号为i的课程目标对应第k个考核环节的权重。

2.课程总目标达成度计算

式中：D——课程总目标达成度;

Di——编号为i的课程分目标对应的达成度；

Qi----编号为i的课程目标的权重。

**九、教材、参考书目及学习资料**

1.拟使用教材

《Python语言程序设计基础》崇天等著，北京·高等教育出版社，2017

1. 其他参考资料

[1] Eric Matthes·Python编程从入门到实践·北京：人民邮电出版社，2016

[2] 何海群·零起点Python大数据与量化交易·北京：电子工业出版社，2017

[3] Yuxing Yan· Python金融实战·北京：人民邮电出版社，2017

[4] 王晓华·Python量化交易实战·北京：清华大学出版社，2019

3. 中国大学MOOC：Python程序设计