

全过程非标准化考核改革

校级优秀案例申报材料

课程名称及代码:	[52018CC106]Python 语言程序设计
所属学院/专业团队:	应用技术学院/人工智能技术系
课程负责人:	张湘怡
主要参与教师:	陈亮、贾宁、金玮、陈秀寓
面向年级/专业:	2025 级/大数据技术、人工智能技术
课程学分/学时:	4 学分/64 学时

改革亮点 (推荐理由)	课程以思政融合、产教协同、数智赋能、评价创新为核心，构建了教学全流程的非标准化考核体系。在数智赋能方面，系统引入 AI 技术辅助命题、教学案例生成与个性化学习支持，推动考核向能力生成转型。通过 AI+真实项目驱动、AI 赋能量规评价、AI 增强学习反馈，显著提升了学生的工程实践能力、创新思维与智能素养，实现了知识、能力与价值塑造的深度融合。		
课程自评 得分	90 分	院级评审 得分	90 分

日期: 2026 年 1 月 12 日

一、改革背景与问题导向

(一) 课程原有考核痛点

《Python 语言程序设计》作为大一专业必修课，旨在培养学生掌握基础语法、序列操作、函数设计等核心知识，并初步形成问题分析、代码实现与项目调试的工程化思维。然而，改革前的课程考核存在以下突出问题：

(1) 命题侧重知识记忆：课改前的考核以闭卷笔试为主，试题多围绕语法细节与固定题型，缺乏对高阶能力的系统性考查。

(2) 考核场景脱离真实应用：题目缺乏源自行业实践或社会需求的真实场景与数据，学生难以建立学习内容与未来岗位的关联认知。

(3) 评价标准导向不清晰：评分缺乏对代码规范性、逻辑严谨性、模块化设计以及文档撰写等工程素养的明确量规，不利于学生形成严谨的编程习惯。

(4) 过程评价薄弱：考核重心偏于期末，形成性评价手段单一、反馈滞后，导致部分学生平时投入不足、期末突击应试，代码实践能力成长缓慢，项目成果同质化现象明显。

上述问题使得原有考核模式难以全面支撑课程核心能力目标的达成，制约了学生工程实践能力与创新思维的培养。

(二) 改革目标

针对传统考核模式的局限，本次课程改革确立了以下四方面具体目标：

(1) 强化高阶能力考核：显著提升非标准化、综合性试题在考核中的比重，引入 AI 辅助命题与多模态场景生成机制，旨在将终结性考核中对复杂问题解决、逻辑思维与创新设计等高阶能力的考查比例提升。

(2) 推动考核场景真实化：深化产教融合与课程思政建设，借助 AI 技术构建行业仿真场景与社会议题任务库，使考核任务贴近真实工程场景与社会需求，有效考查学生的知识迁移与实践应用能力。

(3) 构建全过程、可量化的评价体系：建立并完善贯穿教学全周期的形成性与终结性考核量规，实现评价标准前置化、透明化，通过对学习过程与项目成果的多维度精准评价，促进学生编程规范、工程素养与自主学习能力的阶梯式成长。

(4) 实现学习成果的价值转化：通过改革驱动，引导学生合理使用 AI 工具进行项目设计与创新迭代，提升学生项目成果的创新性、完整性与潜在应用价值，支撑学生在科竞赛与创新实践活动中的表现。

二、考核改革设计实施

针对传统考核痛点，课程以“思政融合、产教协同、数智赋能、评价创新”为核心，对考核体系进行了系统性重构，具体设计与实施如下：

(一) 双维度考核体系强化过程评价

课程建立形成性考核与终结性考核并重的双维度体系。形成性考核贯穿教学全过程，涵盖课堂表现、课后作业、单元测验等环节，强调学习过程的积累。终结性考核采用“闭卷笔试（70%）+项目大作业（30%）”组合，笔试侧重知识整合与逻辑思维，大作业聚焦工程实践与创新应用。

（二）命题贴近真实场景

在期末考试中，将非标准化、综合性试题比例提升，减少对语法点的记忆性考查，增加综合应用题。大作业命题融合企业合作与课程思政，引入“中华经典名句积累”、“党史小知识测试”、“垃圾分类查询工具”等来源于真实需求或具有社会价值的微型项目，要求学生完成从需求分析、代码实现到文档撰写的全流程。

（三）明确评价标准与能力维度

针对各项考核任务，制定详细量规。形成性考核明确各环节评分点；笔试评分标准突出解题思路与代码逻辑；大作业量规则从功能完整性、代码质量、文档规范性、创新性及价值导向等多维度进行分级评价。

（四）深化以评促学

将考核任务嵌入日常教学。通过东软智慧教育平台与智慧职教 MOOC 平台，发布阶梯式学习任务与拓展资源，提供个性化学习支持与实时反馈。在教学过程中开展多次阶段性检查、指导与交流。

（五）AI 赋能非标准化考核全流程

①**AI 辅助命题与场景生成：**利用 AI 工具辅助生成非标准化试题和真实项目场景。

②**AI 驱动的教学案例构建：**在课堂教学中，引入 AI 生成代码案例，展示同一问题的多种解法，引导学生对比分析、优化代码。

③**AI 支持个性化学习与过程反馈：**依托东软智慧教育平台等，集成 AI 智能学情分析系统，为学生提供 24 小时在线的学习支持与代码调试建议。

④**AI 赋能量规评价：**在项目大作业评价中，使用 AI 工具辅助检查代码规范性、逻辑重复率与创新点识别。通过设定 AI 使用报告环节，引导学生合理利用 AI 进行需求分析、代码优化与文档撰写，并在评分细则中明确 AI 工具的合规使用边界。

三、考核改革成效亮点

（一）量化与质性成效

（1）量化成效

①终结性考核中，非标准化、综合应用题占比稳定在 60%以上，其中约 40%的题目借助 AI 辅助生成，实现了对高阶思维能力的重点考查。

②近期课程中，源自企业合作或融入思政元素的大作业选题占比约 30%，如“红色文化问答系统”、“环保垃圾分类助手”等，为学生提供了真实的项目演练场景。

③课程在智慧职教 MOOC 平台已开设至第 4 期，累计选课 312 人，建设并共享课程资源 56 项，拓展了教学覆盖面与影响力。

④改革实施以来，直接支撑学生参与学科竞赛，体现了学习成果的有效转化。

（2）质性提升

①通过大作业训练，学生的需求分析、模块化设计、文档撰写及利用 AI 工具辅助开发的工程实践能力得到系统培养。

②建立了前置、透明、多维的量规评价体系，教师评价有据可依，学生努力方向明确。同时，通过明确 AI 工具使用规范，有效引导了工具的正确使用，防范了学术不端。

③教学融合思政教育，学生在完成技术项目的同时，通过 AI 生成的文化、社会、环保等主题场景，自然深化了对数据伦理、工匠精神、家国情怀等价值理念的认同。

（二）改革前后对比

（1）考核结构重心

改革前：期末笔试定成绩，形成性考核占比低、形式化，导致学生平时松懈、期末突击。

改革后：形成性考核(50%)与终结性考核(50%)并重。形成性考核贯穿全周期；终结性考核采用“闭卷笔试(70%)+项目大作业(30%)”组合。以下为改革后的课程考核与评价：

①课程总成绩构成

课程总成绩（100 分）=形成性考核成绩（50 分×100%）+终结性考核成绩（100 分×50%）

②形成性考核与评价

考核项目	满分值	促进的预期学习效果	评价标准
考勤		具有更新知识和自我完善的学习欲望和良好的学习习惯；	迟到一次扣 1 分，旷课 1 学时扣 2 分；使用考勤系统
课堂表现		具有更新知识和自我完善的学习欲望和良好的学习习惯；	迟到早退、旷课、不遵守上课纪律的扣 1 分；课堂上积极回答问题、帮助同学解决问题的加 1 分。
作业	20	具有更新知识和自我完善的学习欲望和良好的学习习惯； 掌握计算机基础理论知识； 能够应用逻辑思维和实证思维等知识分析、识别、表达复杂工程中的问题； 能够运用专业知识，选择恰当的技术手段处理数据，并对结果进行合理评价； 能够运用大数据、人工智能相关技术原理和方法，分析和评价解决方案的优势和劣势，获得有效结论	4 次作业，每次 5 分，共 20 分；程序运行正确，有清晰注释，代码书写规范，独立完成。
课堂测验	30	掌握计算机基础理论知识； 能够运用专业知识，选择恰当的技术手段处理数据，并对结果进行合理评价； 能够运用大数据、人工智能相关技术原理和方法，分析和评价解决方案的优势和劣势，获得有效结论	三次课堂测验，每次 10 分

③终结考核与评价

终结性考核方式 1:闭卷笔试

考核时长: 90 分钟

预期学习效果	知识点（群）与技能点（能力）	对应掌握程度	是否重点	难中易	所占比例	备注
具有更新知识和自我完善的学习欲望和良好的学习习惯;	Python 语言背景知识和作者介绍	1. 识记 (Remember)	否	易	20.0%	
	Python 代码编写规范以及模块的导入和使用	2. 理解 (Understand)	否	易		
	Python 基于值的内存管理方式	2. 理解 (Understand)	是	难		
	Python 变量和运算符等知识	2. 理解 (Understand)	是	易		
	Python 的基本数据类型	2. 理解 (Understand)	是	易		
	Python 的基础语法规则	2. 理解 (Understand)	是	中		
	Python 的安装与使用	3. 运用 (Apply)	是	中		
掌握计算机基础理论知识;	列表的概念和基本操作	2. 理解 (Understand)	是	中	50.0%	
	序列的基本概念	2. 理解 (Understand)	否	易		
	字符串格式化方法和字符串常用方法	3. 运用 (Apply)	是	易		
	字典与集合的概念和基本操作	3. 运用 (Apply)	是	易		
	元组的概念和基本操作和序列解包	3. 运用 (Apply)	是	易		
	列表推导式	3. 运用 (Apply)	是	中		
	流程控制语句（选择与循环）	3. 运用 (Apply)	是	中		
能够应用逻辑思维和实证思维等知识分析、识别、表达复杂工程中的问题;	切片操作和列表排序	3. 运用 (Apply)	是	中	10.0%	
	函数的定义与使用	2. 理解 (Understand)	是	中		
	函数的形参与实参的概念	3. 运用 (Apply)	是	易		
能够运用专业知识，选择恰当的技术手段处理数据，并对结果进行合理评价；	变量作用域的概念和注意事项	3. 运用 (Apply)	否	中	10.0%	
	参数传递的序列解包	3. 运用 (Apply)	是	难		
	函数基本的参数类型	3. 运用 (Apply)	是	易		
能够运用大数据、人工智能相关技术原理和方法，分析和评价解决方案的优势和劣势，获得有效结论	变量作用域的概念和注意事项	3. 运用 (Apply)	否	中	10.0%	
	参数传递的序列解包	3. 运用 (Apply)	是	难		
	函数基本的参数类型	3. 运用 (Apply)	是	易		

预期学习效果	知识点(群)与技能点(能力)	对应掌握程度	是否重点	难中易	所占比例	备注
具有更新知识和自我完善的学习欲望和良好的学习习惯;	Python 语言背景知识和作者介绍	1. 识记(Remember)	否	易	20.0%	
	Python 代码编写规范以及模块的导入和使用	2. 理解(Understand)	否	易		
	Python 基于值的内存管理方式	2. 理解(Understand)	是	难		
	Python 变量和运算符等知识	2. 理解(Understand)	是	易		
	Python 的基本数据类型	2. 理解(Understand)	是	易		
	Python 的基础语法规则	2. 理解(Understand)	是	中		
	Python 的安装与使用	3. 运用(Apply)	是	中		
掌握计算机基础理论知识;	列表的概念和基本操作	2. 理解(Understand)	是	中	20.0%	
	序列的基本概念	2. 理解(Understand)	否	易		
	字符串格式化方法和字符串常用方法	3. 运用(Apply)	是	易		
	字典与集合的概念和基本操作	3. 运用(Apply)	是	易		
	元组的概念和基本操作和序列解包	3. 运用(Apply)	是	易		
	列表推导式	3. 运用(Apply)	是	中		
	流程控制语句(选择与循环)	3. 运用(Apply)	是	中		
能够应用逻辑思维和实证思维等 知识分析、识别、表达复杂工程 中的问题;	切片操作和列表排序	3. 运用(Apply)	是	中	30.0%	
	函数的定义与使用	2. 理解(Understand)	是	中		
	函数的形参与实参的概念	3. 运用(Apply)	是	易		
能够运用专业知识, 选择恰当的 技术手段处理数据, 并对结果进 行合理评价;	变量作用域的概念和注意事项	3. 运用(Apply)	否	中	20.0%	
	参数传递的序列解包	3. 运用(Apply)	是	难		
	函数基本的参数类型	3. 运用(Apply)	是	易		
能够运用大数据、人工智能相关 技术原理和方法, 分析和评价解 决方案的优势和劣势, 获得有效 结论	变量作用域的概念和注意事项	3. 运用(Apply)	否	中	10.0%	
	参数传递的序列解包	3. 运用(Apply)	是	难		
	函数基本的参数类型	3. 运用(Apply)	是	易		

(2) AI 赋能命题场景真实化

改革前: 命题侧重知识记忆与孤立技能, 多为语法填空、简单编程题, 与产业脱节。

改革后: 笔试命题中非标准化、综合性应用占比 $\geq 60\%$ 。通过 AI 辅助命题, 教师命题时间平均缩短 40%, 试题库动态更新率提高 50%, 题目场景更贴近行业真实需求。大作业选题包含源于校企融合案例或包含课程思政主题。以下为大作业选题案例, 以及学生选题表的部分截图。

序号	选题名称	功能描述（包括但不限于下述功能）	学号	姓名
1	红色景点知识问答	随机出 5 道红色景点题，判断对错并统计得分，附景点背后的革命故事		
2	环保垃圾分类助手	输入垃圾名称，判断类别并统计数量，输出“垃圾分类，绿色生活”提示	25107410111	朱乐超
3	诚信考试承诺书生成器	输入姓名学号，自动生成含“不作弊、守纪律”条款的承诺书文本	25107400102	李浩疆
4	党史小知识测试	10 道党史基础题，计算正确率，正确率 80% 以上显示“继续加油，学史明志”	25107410108	田晓棠
5	校园文明行为统计	记录“主动让座”“随手关灯”等行为次数，生成“文明之星”表扬榜		
6	传统节日习俗查询	输入节日名称，输出日期及习俗，如“端午节：纪念屈原，传承家国情怀”	25107400107	周星佐
7	志愿服务时长记录	用列表存储服务时间，计算总时长，附“奉献社会，青春无悔”鼓励语	25107410105	于希泽
8	中华经典名句积累	存储 10 句经典名言，随机抽取 1 句并解析含义，如“天下兴亡，匹夫有责”	25107410102	张凯皓
9	简易成绩计算器	输入 3 门成绩，计算平均分并判断及格情况	25107410103	翟博文
10	体重 BMI 计算器	输入身高体重，计算 BMI 并提示健康建议	25107410101	梁帅
11	超市购物清单	用列表记录商品名和单价，计算总价	25107410109	徐钰涵
12	猜数字小游戏	系统随机生成 1-50 的数，用户猜并提示“大了 / 小了”	25107410107	王一竹
13	个人日程表	添加 / 查看当天计划，支持按时间排序	25107400112	高雨竹
14	整数加减练习	随机生成 10 以内加减题，统计正确率	25107400105	原野
15	学生点名系统	输入班级名单，随机抽查 1 名学生		
16	字符串反转工具	输入文字，输出反转后的结果，如“上海”→“海上”	24107400323	于天浩
17	简易记账本	记录收入 / 支出金额，计算月末余额	25107400113	姜芸熙
18	月份天数查询	输入年份和月份，输出当月天数	25107400101	刘容成
19	字母大小写转换	输入字符串，一键转为全大写或全小写		
20	5 人成绩排名	输入分数，按从高到低排序并显示名次	25107410106	赵鑫洋

《Python 语言程序设计》大作业选题及代码案例

选题 4：垃圾分类查询与科普工具

选题 1：红色文化知识问答小程序

题目要求

设计一个包含党史、红色文化相关题目的交互式问答程序，实现“出题→用户答题→判断对错→计分→显示成绩和总分”的完整流程。最后输出答题正确率，引导学生了解红色历史。

覆盖课标知识点

- KT4 (基本数据类型：字符串、整数)、KT5 (变量与运算符：计分变量、比较运算符)
- KT9 (列表：存储题目集合)、KT14 (字典：存储单题的“题干 - 选项 - 答案”)
- KT11 (流程控制：for 循环答题、if 判断答案)
- KT15 (字符串格式化：题干与选项展示)
- KT16 (函数：答题逻辑封装)、KT17 (函数形参 / 实参)

代码案例

```
def red_culture_quiz():
    # 1. 用列表存储题目。每个题目是字典（包含题干、选项、正确答案）
    questions = [
        {
            "question": "中国共产党成立的时间是？",
            "options": ["A. 1919 年", "B. 1921 年", "C. 1949 年", "D. 1978 年"],
            "answer": "B"
        },
        {
            "question": "红军长征胜利结束的标志是？",
            "options": ["A. 遵义会议", "B. 吴起镇会师", "C. 会宁会师", "D. 飞夺泸定桥"],
            "answer": "C"
        }
    ]
```

题目要求

存储 20 种常见垃圾的“名称→类别（可回收/有害/厨余/其他）→科普知识《如危害/处理方式》”，支持用户输入垃圾名称查询（忽略大小写），统计查询次数，最后显示“查询最多的垃圾”，培养学生环保意识。

覆盖课标知识点

- KT4 (基本数据类型：字符串)、KT5 (变量与运算符：计数)
- KT14 (字典：存储垃圾“名称-类别-科普”、查询次数统计)
- KT11 (流程控制：循环查询、if 判断垃圾是否存在)
- KT15 (字符串操作：忽略大小写、科普内容展示)、KT16 (函数：查询、统计)

代码案例

```
def init_garbage_dict():
    # 初始化垃圾字典 (KT14: 字典嵌套)
    return {
        "电池": {
            "type": "有害垃圾",
            "intro": "含有汞、镉等有毒物质，随意丢弃会污染土壤和水源，需投入有害垃圾专用箱。"
        },
        "塑料瓶": {
            "type": "可回收垃圾",
            "intro": "材质为 PET，可回收再加工成纤维、塑料容器等，投放前建议压扁节省空间。"
        }
    }
```

(3) AI 驱动案例动态生成

改革前：教师授课以静态讲解为主，学生学习路径线性固定，通过观察静态代码和听取口头描述来理解编程逻辑。

改革后：通过“飞象老师”等AI工具，将抽象代码逻辑转化为逐步推进、多维度联动的可视化动画。在讲解如Python函数参数类型这类难点时，AI工具生成交互式动画网页，使学生能直观、连贯地理解参数传递全过程。教师可一键生成分层案例，针对不同学生提供基础、进阶、综合等不同难度的动画任务。引入AI动画后，学生理解复杂知识点的平均时间缩短，课堂互动频率有所提升，前排就坐率较高。下图展示的是部分课程画像，以及部分AI生成的代码案例。



1. 位置参数 (Positional Arguments)

最基础的参数类型，实参和形参按位置一一对应

核心概念

- 调用时实参和形参的顺序必须严格一致
- 实参和形参的数量必须相同
- 这是最常用的参数传递方式
- 就像团队协作，每个人各司其职，位置不能乱

进阶示例：数学运算

```
def calculate(x, y, operator):  
    if operator == '+':  
        return x + y  
    elif operator == '-':  
        return x - y  
  
# 计算 10 + 5  
result = calculate(10, 5, '+')  
print(result)
```

x=10 → y=5 → operator=+

✓ 输出结果:
15

入门示例：打印三个参数

```
def demo(a, b, c):  
    print(a, b, c)  
  
# 正确调用: 按位置传递  
demo(3, 4, 5)
```

a=3 → b=4 → c=5

✓ 输出结果:
3 4 5

顺序很重要

```
def demo(a, b, c):  
    print(a, b, c)  
  
# 改变顺序, 结果不同  
demo(3, 5, 4)
```

a=3 → b=5 → c=4

X 常见错误：参数数量不匹配

```
def demo(a, b, c):
    print(a, b, c)

# 错误：传递了4个参数，但函数只需要3个
demo(1, 2, 3, 4)
```

X 错误信息：

TypeError: demo() takes 3 positional arguments but 4 were given

X 常见错误：参数顺序错误

```
def divide(dividend, divisor):
    return dividend / divisor

# 错误：把除数和被除数位置搞反了
# 想计算  $10 / 2 = 5$ , 但写成了  $2 / 10 = 0.2$ 
result = divide(2, 10)
print(result)
```

dividend=2(错) → divisor=10(错)

✓ 输出结果：
0.2 # 结果错误！应该是5

案例1：整数传递

案例2：字符串传递

案例1：整数传递 - 成绩修改

```
# 不可变对象：整数
def try_modify_score(score):
    print(f"函数内修改前: score = {score}")
    score = score + 10 # 创建新对象
    print(f"函数内修改后: score = {score}")

# 实参
original_score = 85
print("调用前: original_score = {original_score}")

try_modify_score(original_score)

print("调用后: original_score = {original_score}")
# 输出仍是85，未被修改！
```

原理解析

- original_score 指向整数对象 85
- 调用函数时，score 也指向同一个 85
- score = score + 10 创建新对象 95
- score 指向新对象，original_score 仍指向 85

总结与对比

核心要点

Python中的参数传递本质是“引用传递”，形参和实参指向同一个对象。是否影响实参取决于：对象是否可变 以及 是修改内容还是重新赋值。

完整对比表

对象类型	操作方式	是否影响实参	示例代码
不可变对象 (int, str, tuple)	重新赋值	✗ 不影响	x = x + 1 s = s + "abc"
可变对象 (list, dict, set)	修改内容	✓ 影响	lst.append(x) lst[0] = x dict['key'] = value
	重新赋值	✗ 不影响	lst = [1, 2, 3] lst = lst + [4]

不可变对象特点

- 对象创建后内容不能修改
- 任何“修改”操作都会创建新对象
- 函数内修改形参不影响实参
- 包括：int, float, str, tuple, bool

内存图解

1. 调用前



2. 函数调用时



指向同一个对象

3. 函数内修改后



指向不同对象

结论

函数内对形参的修改不影响实参，因为整数是不可变对象，重新赋值会创建新对象。

(4) 新增大作业考核方式与评价标准

改革前：期末考试代表全部的终结性评价。

改革后：终结性加入大作业考核，从功能完整性、代码质量、文档规范性、创新性、价值导向等个维度进行分级评价，同时明确 AI 辅助工具的使用规范与边界。下表为新增的大作业评分细则。

评分项目	分值	评分细则
1. 选题功能完成度	10	完全实现所选题目全部核心功能，逻辑连贯、贴合实际需求，得 8-10 分； 实现 80%以上核心功能，仅少量细节缺失，得 6-7 分； 实现 60%-80%核心功能，关键功能有缺失，得 4-5 分； 实现 60%以下核心功能，得 0-3 分。
2. 变量与输入输出规范	8	变量命名规范、逻辑合理， <code>input()</code> 含清晰引导提示， <code>print()</code> 输出格式整洁准确，得 7-8 分； 变量命名较规范，输入提示简略或输出格式稍乱，得 5-6 分； 变量命名无意义，或无输入提示、输出混乱，得 3-4 分； 未用 <code>input()</code> 动态输入，得 0-2 分。
3. 序列操作准确性	8	熟练使用列表/字符串等序列，准确运用索引、切片等操作，无逻辑错误，得 7-8 分； 使用序列但操作存在少量问题，得 5-6 分； 仅定义序列未操作，或操作完全错误，得 3-4 分； 未使用任何序列，得 0-2 分。
4. 条件判断 <code>if-else</code> 逻辑	8	合理使用 <code>if-elif-else</code> 结构，逻辑无漏洞，得 7-8 分； 条件判断逻辑基本正确，但未覆盖 1-2 个边界值，得 5-6 分； 条件判断存在明显逻辑错误，得 3-4 分； 未使用 <code>if-else</code> ，得 0-2 分。
5. 循环结构合理性	8	用 <code>for/while</code> 循环高效处理重复操作，循环终止条件清晰无死循环，得 7-8 分； 循环能实现功能，但终止条件冗余或循环次数偏差 1-2 次，得 5-6 分； 循环存在死循环风险，或仅能执行 1 次，得 3-4 分； 未使用循环，得 0-2 分。
6. 函数设计与调用	10	定义 3 个及以上功能独立的函数，调用逻辑清晰、无冗余代码，得 8-10 分； 定义 2 个函数且调用正常，或 3 个以上函数但有少量冗余，得 6-7 分； 定义 1 个函数且调用无错，或 3 个以上函数但调用逻辑混乱，得 4-5 分； 定义函数调用错误，得 2-3 分； 未定义函数，得 0-1 分。
7. 代码健壮性	8	代码运行正常，无崩溃风险，得 5-8 分； 代码运行有明显风险，得 0-4 分。
8. 代码注释	8	代码注释覆盖函数功能、关键逻辑，得 7-8 分； 无语法错误但注释缺失，得 5-6 分； 代码无注释，得 0-4 分。
9. 报告结构	7	包含“项目背景与需求、核心功能设计、核心代码说明、运行截图、总结反思”5 部分，得 6-7 分；

完整性		缺失 1 部分, 得 4-5 分; 缺失 2 部分, 得 2-3 分; 缺失 3 部分及以上, 得 0-1 分。
10. 报告内容深度	7	核心代码说明含逻辑拆解、函数作用解释, 总结反思含问题解决过程、AI 使用心得, 得 6-7 分; 代码说明简略, 总结无反思, 得 4-5 分; 内容拼凑、逻辑混乱, 得 0-3 分。
11. 报告图文规范性	8	截图清晰, 无错别字, 格式统一, 得 5-8 分; 截图模糊或有 1-2 处错别字, 得 3-4 分; 截图缺失或 3 处以上错别字, 得 1-2 分; 无截图且错漏多, 得 0 分。
12. AI 工具使用与应用能力	10	明确 AI 工, 描述 AI 在需求分析、代码调试、功能优化等 3 个及以上环节的应用, 能识别并修正 AI 生成的逻辑漏洞, 且反思 AI 局限性, 得 8-10 分; 说明 AI 工具及 2 个环节应用, 能修正明显错误, 无反思, 得 6-7 分; 提及 AI 工具及 1 个环节应用, 未修正错误, 得 4-5 分; 仅说用 AI 无具体环节, 得 2-3 分; 未用 AI 或未提及, 得 0-1 分。
总分	100	——

(5) 数字化立体课堂建设

改革前：教学与考核相对脱节，难以支撑能力阶梯式成长。

改革后：依托东软智慧教育平台实现作业发布、AI 答疑、学情分析；课程中充分发挥 NeuAI 功能群的技术优势，打造“教、学、评、辅”全链路数智化场景。同时，利用智慧职教 MOOC 提供多项拓展资源，支撑个性化学习。下图展示的是智慧教育平台构建的围绕课堂教学需求搭建资源与互动体系与 MOOC 平台的课程相关情况。

深耕专业知识，打造高校专属智库
整合AI知识库，用智能手段开启知识管理新旅程

列表切片的语法是: `object[start_index:end_index:step]`。

- `start_index`: 表示起始索引(包含该索引对应值); 该参数省略时, 表示从对象“端点”开始取值, 至于是从“起点”还是从“终点”开始, 则由 `step` 参数的正负决定, `step` 为正从“起点”开始, 为负从“终点”开始。
- `end_index`: 表示终止索引(不包含该索引对应值); 该参数省略时, 表示一直取到数据“端点”, 至于是到“起点”还是到“终点”, 同样由 `step` 参数的正负决定, `step` 为正时直到“终点”, 为负时直到“起点”。
- `step`: 正负数均可, 其绝对值大小决定了切取数据时的“步长”, 而正负号决定了“切取方向”, 正表示“从左往右”取值, 负表示“从右往左”取值。当 `step` 省略时, 默认为1, 即从左往右以步长1取值。

一个完整的切片表达式包含两个冒号, 用于分隔三个参数。第一个数字表示切片开始位置(默认为0); 第二个数字表示切片截止(但不包含)位置(默认为列表长度); 第三个数字表示切片的步长(默认为1), 当步长省略时可以顺便省略最后一个冒号。
当只有一个“:”时, 默认第三个参数 `step = 1`; 当一个“:”也没有时, 表示切取 `start_index` 指定的那个元素, 相当于取指定位置的值。

此回答基于以下知识库资料作为参考:

- 6第2章 列表推导式和迭代器与可迭代对象.pptx
- 5第3章 选择与循环-作业2.pptx

空间容量
22.22MB/2GB

输入内容/选择技能

AI也可能犯错。请核查重要信息

Python语言程序设计

层次: 高职专科 专业类别: 电子与信息大类

Python语言作为当下最火爆的语言之一,具有简单学习成本低、功能强大、代码库丰富等特点。在大数据、人工智能、自动化运维等众多领域有着广泛的应用。对于已经掌握了基本编程理论的同学来说, Python是一门很好的补充语言,它能够培养学生编程思想,提高学生的编程技能,提升...

陈亮 大连东软信息学院

2025年10月09日 - 2026年01月31日 648时

去学习 ▶

已开课 第4期开课 312人 累计选课人数 本期16人 21个 学员所属单位 本期5个 2次 累计互动次数 本期0次 1622条 累计日志总数

教学团队 课程详情 课程大纲 考核标准 课程评价

课程负责人

陈亮 讲师 大连东软信息学院

毕业于吉林大学,硕士研究生,4年大数据项目开发和管理经验,曾就职于京东、阿里和知乎三家互联网企业。目前就职于大连东软信息学院,高等职业技术学院人工智能技术系...

教师团队

贾宁 大连东软信息学院 张湘怡 大连东软信息学院

相关课程推荐

- 信息技术应用 成都航空职业技术大学 4391人参加
- 人工智能应用导论 广东科学技术职业学院 2330人参加
- 云计算平台技术与应用 武汉船舶职业技术学院 1406人参加
- 信息技术 1296人参加

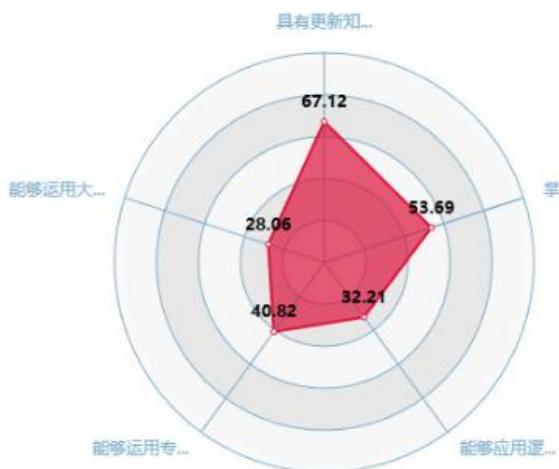
客服 帮助中心

(6) 课程评估指标

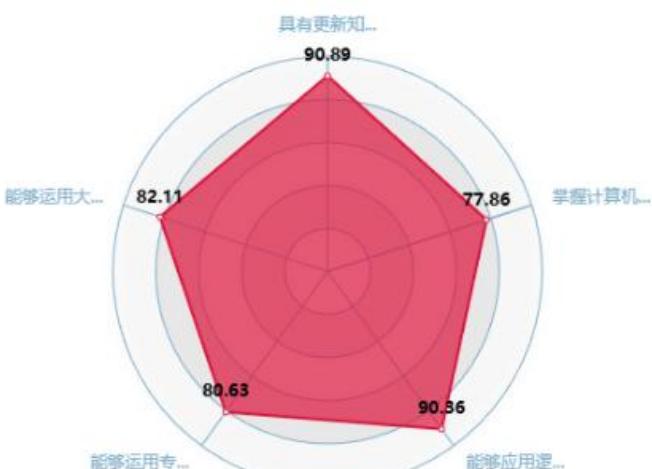
改革前: 在传统考核模式下,课程各项预期学习效果的达成度整体处于中低水平。

改革后: 课程改革实施后,通过全过程非标准化考核体系的系统构建与AI赋能教学的深入应用,学生在各项预期学习效果上的达成度实现显著提升,尤其在自主学习能力、复杂问题分析能力、专业技术实践能力等关键维度进步明显。下图左侧数据反映改革前各项能力指标的达成表现,右侧数据则对应改革后的评估结果,两者对比可以直观看出课程改革带来的显著提升。

各个课程预期学习效果平均达成度



各个课程预期学习效果平均达成度



(7) 改革前：在传统考核模式下，23、24年的课程成绩结构存在明显问题。总成绩及格率偏低，形成性考核和终结性考核差值过大，成绩分布不合理。

学年	学生数	形成性考核成绩及格率	终结性考核成绩及格率	总成绩及格率
2024	157	89.00%	31.00%	60.00%
2023	180	98.00%	11.00%	66.00%

改革后：形成性考核与终结性考核及格率均稳定在 75%，体现学习过程与考核结果的一致性。总成绩及格率提升至 80%，平均分达到 74.85 分，成绩分布显著改善：不及格比例控制在 20%，中等分数段（70-89 分）占比达 55%，优秀段（90-100 分）学生比例达 20%，最高分达 97 分。

形成性成绩分布情况 (无学习过程不在统计范围内)						
教学班号	任课教师	平均分	及格率	及格以下 (30分以下)	及格以上 (30分以上)	满分 (50分)
320007-001	张湘怡	34.03	75.00%	25.00%	70.83%	4.17%
总计		34.03	75.00%	25.00%	70.83%	4.17%

终结性成绩分布情况 (无学习过程、缺考、缓考、取消资格、舞弊不在统计范围内)								
教学班号	任课教师	平均分	及格率	< 60	60-69	70-79	80-89	90-100
320007-001	张湘怡	68.23	75.00%	25.00%	20.00%	20.00%	20.00%	15.00%
总计		68.23	75.00%	25.00%	20.00%	20.00%	20.00%	15.00%
最高分	97.00			最低分		18.10		

综合成绩分布情况 (无学习过程、缺考、缓考、取消资格、舞弊不在统计范围内)								
综合成绩构成	综合成绩(百分制)=平时成绩(百分制)(100.0%)+末考成绩(百分制)(50.0%)							
教学班号	任课教师	平均分	及格率	< 60	60-69	70-79	80-89	90-100
320007-001	张湘怡	74.85	80.00%	20.00%	5.00%	30.00%	25.00%	20.00%
总计		74.85	80.00%	20.00%	5.00%	30.00%	25.00%	20.00%

(8) 学习成果与产出

改革前：课程产出应用价值低，多为习题复现，难以体现创新与解决实际问题的潜力。

改革后：大作业成果具备初级项目形态，经过后期课程迭代，优秀成果可转化为学科竞赛等。以下为部分学生竞赛获奖情况。

竞赛名称	获奖等级	获奖年份
第二十七届中国机器人及人工智能大赛“海光杯”辽宁赛区选拔赛	省级三等奖	2025
中国国际大学生创新大赛	校级银奖	2025
第二届大学生高校物理挑战赛	二等奖	2025
第五届 ICAD 国际当代青年美术设计大赛	银奖	2025
第八届 EPACC 国际环保公益设计大赛	金奖	2025
2024 全国大学生高新技术竞赛	Word 赛道一等奖	2025

(三) 学生优秀作答/成果

(1) 大作业案例

“中华经典名句积累”，部分项目报告截图如下：

《Python 语言程序设计》课程大作业-项目报告

姓名:	张凯皓	学号: 20107410102
科目	Python 语言程序设计	
项目名称	中华经典名句积累	

一、项目简介
本项目主题为“中华经典名句积累”，核心功能是存储 10 句中华经典名言，并实现随机抽取名言、展示名言解析的交互功能。
选择该主题的原因：一是中华经典名言是传统文化的精华，通过程序积累有助于学习；二是项目需求能覆盖 Python 课程的核心知识点（变量、序列、条件判断、循环、函数），适合作为课程大作业实践。

二、项目核心代码说明

(1) 变量与输入输出
实现：通过 `input()` 获取用户的操作选择（如“是否查看解析”），通过 `print()` 输出名言、解析及提示信息。
核心代码示例：
`# 获取用户输入
user_choice = input("是否查看这句名言的解析？(输入 Y 查看, 输入 N 跳过, 输入 Q 退出): ")`
`# 输出结果
print(f"\u25aa 随机抽取的经典名言: {selected_quote}")`

(2) 序列的使用
实现：使用列表存储 10 句名言及对应解析（每个列表元素是包含“名言 + 解析”的子列表），通过索引操作抽取名言。

四、总结

(1) 收获
掌握了 Python 核心语法（变量、列表、条件判断、循环、函数）的实际应用；学会了将功能拆分到不同函数中，提升代码的模块化和可读性；理解了用户交互逻辑的设计思路，能通过 `input()` 和 `print()` 实现简单的人机交互。

(2) 遇到的困难及解决方法
困难：随机抽取名言时，无法正确通过索引获取子列表中的名言 / 解析。
解决方法：
通过 `len(quotes_list)-1` 确定索引范围，并用 `quotes_list[random_index][0]`、`quotes_list[random_index][1]` 的双层索引实现取值。

(3) 对 Python 编程的理解
Python 是一门简洁、易读的语言，通过“函数 + 模块化”的方式可以将复杂功能拆分为简单步骤；同时，列表、循环、条件判断等基础语法是实现程序逻辑的核心，合理组合这些语法可以完成实用的小工具。

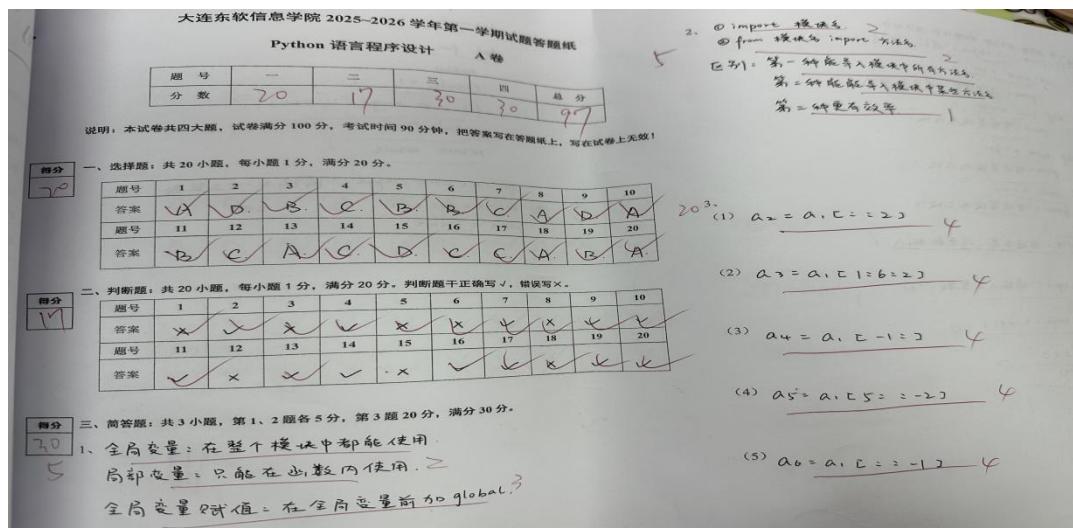
五、AI 工具使用情况
使用了豆包 AI 工具，辅助完成以下工作：提供项目功能设计思路；编写核心代码的框架，并解释代码的技术要点；辅助整理项目报告的内容结构。

该大作业成果是课程考核改革的典型产出，其亮点如下：

- ①基础能力扎实：精准运用列表、条件判断、循环、函数等核心语法，逻辑清晰，实现了完整的交互功能。
- ②高阶思维初显：采用模块化函数设计，体现了初步的系统工程思维和代码组织能力。
- ③实践创新融合：将技术练习与中华传统文化传承有机结合，是“专业教学与课程思政”自然融合的生动范例。
- ④价值延伸可观：项目具备清晰的原型架构，易于扩展为其他主题的问答工具；学生明确并合理使用了 AI 工具辅助开发，体现了数字化时代的学习与创新能力。

(2) 试卷案例

本次课程期末考试中，多位学生获得优异成绩，其中最高分为 97 分。该试卷体现了经过全过程非标准化考核训练后，学生所具备的扎实编程基础、清晰逻辑思维与灵活问题解决能力，是课程改革在知识考查维度上的重要成果例证。



四、总结反思与持续改进

本轮改革通过构建过程与结果并重、真实项目驱动、量规引导评价的体系，学生工程实践能力与创新意识提升明显。未来需进一步深化校企协同命题机制，增强项目选题的前沿性与综合性。

学生感悟：“通过大作业，我不仅学会了把多个知识点串联起来解决真实问题，还第一次感受到自己写的代码可以承载文化传播的价值。”

本课程将以智能化发展为牵引，持续深化 AI 在全过程非标准化考核中的融合应用。下一步，计划建立 AI 命题资源库，强化试题的前沿性与综合性；依托学习行为数据探索适应性评价机制，推动考核路径个性化发展；同时构建 AI 支持的项目孵化平台，促进优秀成果向竞赛与创新实践转化，逐步形成教学与创造的闭环生态，持续提升课程育人实效与示范价值。

附件：除院级评审表外，可附文中未尽的其他相关支撑材料。

附件 1：《[52018CC106]Python 语言程序设计》全过程非标准化考核改革的评分表（院级评审）

一级指标	二级指标	三级指标	指标解读	院评得分	申报材料是否佐证
考核设计(30分)	1.1 目标适配性(10分)	1.1.1 培养目标对齐度	考核任务与课程目标高度一致，覆盖知识、能力、素养三个维度，无单一维度偏废。	9	是
	1.2 场景任务设计真实性(10分)	1.2.1 场景真实性	跳出传统经验命题局限，至少1个任务的背景、数据、问题源于真实行业、社会或前沿案例，贴近实践。	5	是
		1.2.2 命题多元性	建立校内教师+AI+行业专家（至少1名）等多元主体协同命题机制。	4	否
	1.3 命题高阶性(10分)	1.3.1 任务高阶能力导向	记忆类题目≤30%（根据课程目标调整），提升考查学生批判性思维、知识迁移与应用、创新实践、问题解决等高阶能力考查题目占比，问题具有灵活性、探究性和开放性。	9	是
		1.3.2 任务综合性与挑战度	题目具有整合性，难易梯度清晰，能有效区分不同能力水平学生；无超纲或无意义难题，挑战度贴合学生认知水平与课程要求。		是
评价实施(40分)	2.1 评价标准开放性(15分)	2.1.1 指标明确性与可视度	无“千人一面”的标准化要求，建立分级评价量表，每个维度的指标可观察、可考量、可评价、可验证，有效降低评阅人主观偏差，避免主观臆断。		是
		2.1.2 开放性与限制性	高阶试题答案具有开放性（无现成答案可找），尊重学生思维和方法差异性，允许选择不同技术路径、研究方法或成果形式，预留创新空间；需在评价标准中明确AI工具的使用场景、使用边界及违规判定标准，避免AI抄袭。	14	是
		2.1.3 标准公示及时性	评价标准（含评价维度、分级指标、评分权重等）在考核任务启动前向学生完整公示，明确告知“如何评、评什么”，避免学生学习方法偏差以及成果偏离要求，助力实现“评价即学习”。		是
	2.2 过程培养有效性(15分)	2.2.1 教学过程互动性与引导性	教学方法善于融入互动、引导探究、组织讨论，调动学生积极性，激发潜能；学习任务具有阶梯式，引导学生能力逐级提升，杜绝“教学空转、期末叠加难度”断层现象。		是
		2.2.2 过程支持与技术赋能	依托信息化平台、AI等工具，赋能学生自学自评与能力提升，通过过程跟踪、成果答辩核验等方式验证成果真实性，避免成果代做。	14	是
		2.2.3 及时个性化反馈	在学习任务关键节点嵌入精准反馈与及时指导，而非仅给出分数或笼统评语，提供学生个体的具体改进建议和学习支持，解决学生能力短板。		是

一级指标	二级指标	三级指标	指标解读	院评得分	申报材料是否佐证
改革成效 (30分)	2.3 评分公平性(10分)	2.3.1 评分客观公信度	试批试评校准评分尺度；运用AI等技术工具赋能评阅，提升效率与精准度；建立健全多层级评分复核机制，学生成绩复查量低且复查结果零差错。	9	是
		2.3.2 多元评价融合	融合教师评价、学生互评、行业专家点评等多元主体；互评通过制定成果贡献度量化评分项、评价主体回避、异常值筛查等机制规则，避免“搭便车”、印象打分、成果归属或责任划分不清等现象。		是
	3.1 学生能力提升(15分)	3.1.1 学习主动性激发	学生参与过程性互动研讨、主动优化成果的比例显著提升，形成以考促学、以学促能的良性循环。	14	是
		3.1.2 高阶能力达成	以学生作答及成果为核心证据，有效印证学生在理论联系实际的实践应用、跨域知识整合、复杂问题拆解、创新方案设计与实践落地等方面成效，无死记硬背、抄袭拼凑现象及AI作弊行为。		是
	3.2 成果价值(10分)	3.2.1 实践应用价值	学生成果具备明确的问题解决逻辑与实践价值，或具备落地潜力，而非单纯的任务作业。	8	是
		3.2.2 创新性与个性化	成果体现学生独特思考或个性化表达，具有原创突破，无同质化现象。		是
	3.3 持续改进(5分)	3.3.1 教与学优化联动	学生满意度不低于80%，学生自我评估良好；改革有效解决原有考核问题，并基于考核结果调整课程教学策略，实现以考促教的闭环。	4	否
合计得分：				90	-

说明：1. 总分采用100分制，按二级指标细化评分，求和计算最终得分。

2. 自评与评审给分：二级指标按三档评分——“达标”（满分）：完全符合指标要求，核心量化指标达标，无任何偏差、“基本达标”（满分的60%—80%）：符合指标核心要求，次要量化指标存在轻微偏差（不超过10%），无实质性影响；“不达标”（0—59%）：未满足指标核心要求，或触发相关否决项关联条件。

3. 核心否决项：若存在以下情形之一，直接判定为“不达标”（分数不高于59分）：

- ①命题开放性不足（如记忆类题目占比超过30%且未设置开放性或高阶考查题目）；
- ②评价标准缺失，或标准表述模糊、无明确评判依据、无高阶能力评价、不可操作；
- ③过程评价缺乏有效支撑材料证明过程性培养学生能力发展；
- ④学生成果存在抄袭、拼凑行为，或经核查确认存在AI作弊情况。