# Python400集 视频教程

**高淇老师 带你零基础手写神经网络**

本套Python课程400集，从零基础开始，全面讲解Python基础，直达可以手写神经网络的境界。

我们创造性的增加了：手写数据结构和算法、手写神经网络项目。让零基础学员在学习Python时就开始接触高深的底层知识、神经网络底层实现的知识。

内容共分26章，讲解了Python基础的方方面面，可以作为Python学习者的“宝典”。同时，穿插了3个大型实战项目：

1. 坦克大战游戏。   
    采用游戏开发的方式，寓教于乐，快乐中掌握Python基础。
2. 手写算法和数据结构。   
    算法和数据结构是很多大公司面试的不二选择，我们这里讲解内容的同时，手把手  
    带你实现一个个底层算法，将内功练扎实。
3. 手写神经网络。   
    Python是人工智能的第一语言。我们创造性的在基础课程中就加入了如何编写一  
    个自己的神经网络，为踏入神经网络的大门打下“坚实基础”。

本套视频一共400集，本套视频共分3季：

第一季 【基础篇】Python基础 （115集）

第二季 【提高篇】Python深入和扩展（100集）

第三季 【扩展篇】网络编程、多线程、扩展库（85集）

第四季 【高手篇】算法、Python源码、函数式编程、手工实现神经网络（100集）

现公布第一季，欢迎大家下载观看。后续视频，将会陆续公布。

# 第一章 Python入门

001. Python介绍\_特性\_版本问题\_应用范围

002. Python下载\_安装\_配置\_第一行Python程序

003. 开发环境介绍\_交互模式的使用\_IDLE介绍和使用

004. IDLE开发环境的使用\_建立Python源文件

005. Python程序格式\_缩进\_行注释\_段注释

006. 简单错误如何处理\_守破离学习法\_程序员修炼手册

007. 海龟绘图\_坐标系问题\_画笔各种方法

008. 海龟绘图\_画出奥运五环图

# 第二章 编程基本概念

1. 程序的构成

2. 对象的基本组成和内存示意图

3. 引用的本质\_栈内存和堆内存\_内存示意图

4. 标识符\_帮助系统的简单使用\_命名规则

5. 变量的声明\_初始化\_删除变量\_垃圾回收机制

6. 链式赋值\_系列解包赋值\_常量

7. 内置数据类型\_基本算术运算符

8. 整数\_不同进制\_其他类型转换成整数

9. 浮点数\_自动转换\_强制转换\_增强赋值运算符

10. 时间表示\_unix时间点\_毫秒和微秒\_time模块

11. 多点坐标\_绘出折线图\_计算两点距离

12. 布尔值\_比较运算符\_逻辑运算符\_短路问题

13. 同一运算符\_整数缓存问题

14. 字符串\_unicode字符集\_三种创建字符串方式\_len()

15. 字符串\_转义字符\_字符串拼接\_字符串复制\_input()获得键盘输入

16. 字符串\_str()\_使用[]提取字符\_replace()替换生成新字符串\_内存分析

17. 字符串\_切片slice操作\_逆序

18. 字符串\_split()分割\_join()合并\_join()效率测试

19. 字符串\_驻留机制\_内存分析\_字符串同一判断\_值相等判断

20. 字符串\_常用查找方法\_去除首位信息\_大小写转换\_排版

21. 字符串\_format格式化\_数字格式化操作

22. 可变字符串\_io.StringIO

23. 运算符总结\_位操作符\_优先级问题

# 第三章 序列

1. 列表\_特点\_内存分析
2. 创建列表的4种方式\_推导式创建列表
3. 列表\_元素的5种添加方式\_效率问题
4. 列表\_元素删除的三种方式\_删除本质是数组元素拷贝
5. 列表\_元素的访问\_元素出现次数统计\_成员资格判断
6. 列表\_切片slice操作
7. 列表\_排序\_revered逆序\_max\_min\_sum
8. 列表\_二维列表\_表格数据的存储和读取
9. 元组\_特点\_创建的两种方式\_tuple()要点
10. 元组\_元素访问\_计数方法\_切片操作\_成员资格判断\_zip()
11. 元组\_生成器推导式创建元组\_总结
12. 字典\_特点\_4种创建方式\_普通\_dict\_zip\_formkeys
13. 字典\_元素的访问\_键的访问\_值的访问\_键值对的访问
14. 字典\_元素的添加\_修改\_删除
15. 字典\_序列解包用于列表元组字典
16. 字典\_复杂表格数据存储\_列表和字典综合嵌套
17. 字典\_核心底层原理\_内存分析\_存储键值对过程
18. 字典\_核心底层原理\_内存分析\_查找值对象过程
19. 集合\_特点\_创建和删除\_交集并集差集运算

# 第四章 控制语句

1. Pycharm开发环境的下载安装配置\_项目创建和运行

2. 单分支选择结构\_条件表达式详解

3. 双分支选择结构\_三元运算符的使用详解

4. 多分支选择结构

5. 选择结构的嵌套

6. while循环结构\_死循环处理

7. for循环结构\_遍历各种可迭代对象\_range对象

8. 嵌套循环

9. 嵌套循环练习\_九九乘法表\_打印表格数据

10. break语句

11. continue语句

12. else语句

13. 循环代码优化技巧（及其重要）

14. zip()并行迭代

15. 推导式创建序列\_列表推导式\_字典推导式\_集合推导式\_生成器推导式

16. 综合练习\_绘制不同颜色的多个同心圆\_绘制棋盘

# 函数

1. 函数的基本概念\_内存分析\_函数的分类\_定义和调用

2. 形参和实参\_文档字符串\_函数注释

3. 返回值详解

4. 函数也是对象\_内存分析

5. 变量的作用域\_全局变量\_局部变量\_栈帧内存分析讲解

6. 局部变量和全局变量\_效率测试

7. 参数的传递\_传递可变对象\_内存分析

8. 参数的传递\_传递不可变对象\_内存分析

9. 浅拷贝和深拷贝\_内存分析

10.参数的传递\_不可变对象含可变子对象\_内存分析

11.参数的类型\_位置参数\_默认值参数\_命名参数

12.参数的类型\_可变参数\_强制命名参数

13.lambda表达式和匿名函数

14.eval()函数用法

15.递归函数\_函数调用内存分析\_栈帧的创建

16.递归函数\_阶乘计算案例

17.嵌套函数\_内部函数\_数据隐藏

18.nonlocal\_global

19. LEGB规则

# 第六章 面向对象编程

1. 面向对象和面向过程的区别\_执行者思维\_设计者思维

2. 对象的进化故事

3. 类的定义\_类和对象的关系

4. 构造函数\_\_init\_\_

5. 实例属性\_内存分析

6. 实例方法\_内存分析方法调用过程\_dir()\_isinstance

7. 类对象

8. 类属性\_内存分析创建类和对象的底层

9. 类方法\_静态方法\_内存分析图示

10. \_\_del\_\_()析构方法和垃圾回收机制

11. \_\_call\_\_()方法和可调用对象

12. 方法没有重载\_方法的动态性

13. 私有属性

14. 私有方法

15. @property装饰器\_get和set方法

16. 面向对象的三大特征说明(封装、继承、多态)

17. 继承

18. 方法的重写

19. object根类\_dir()

20. 重写\_\_str\_\_()方法

21. 多重继承

22. mro()

23. super()获得父类的定义

24. 多态

25. 特殊方法和运算符重载

26. 特殊属性

27. 对象的浅拷贝和深拷贝\_内存分析

28. 组合

29. 设计模式\_工厂模式实现

30. 设计模式\_单例模式实现

# 第七章 模块

1. 模块化程序设计理念\_模块和包
2. 标准库有哪些
3. 模块的设计和实现
4. API的设计
5. 模块的导入和使用
6. 包的使用\_导入和创建
7. 重新加载模块\_动态代入模块
8. 模块的导入顺序
9. 命名空间和名称查找顺序
10. 第三方扩展库的介绍和安装
11. PyPy
12. PiP安装第三方扩展库

# 第八章 文件操作（IO）

1. 文本文件和二进制文件

2. 文件操作相关模块介绍

3. 创建文件对象\_文本文件写入

4. with语句\_上下文管理

5. 文本文件的读取

6. 二进制文件的读取和写入

7. 使用pickle序列化

8. CSV文件的操作\_csv文件读取

9. CSV文件的操作\_csv文件写入

10. os和os.path模块\_文件级操作

11. os和os.path模块\_目录级操作

12. 存储数据\_json.dump()和json.load()

12. 练习1

13. 练习2

14. 练习3

# 第九章 异常和错误

1. 异常处理结构\_try\_except

2. try\_finally

3. raise抛出异常

4. 自定义异常

5. 常见异常的解决

6. with上下文管理器

7. 程序调试

8. 使用pdb模块调试程序

9. IDLE的调试

10. Pycharm开发环境的调试

# 数值日期和复杂文本处理

1. math模块
2. random模块\_随机整数\_随机序列
3. 数值运算模块NumPy\_数组
4. 数值运算模块NumPy\_应用
5. 时间表示原理\_时间对象
6. 日期对象
7. 字符串和日期对象互相转换

# 第11章 GUI编程

1. Tkinter介绍

2. 第一个GUI程序

3. 事件处理

4. 控件\_Button按钮

5. 控件\_Canvas画布

6. 控件\_Chekbutton\_Radiobutton

7. 控件\_Entry文本输入框

8. 控件\_Frame容器

9. 控件\_Label\_Message

10. 控件\_菜单制作\_Menu\_Menubutton

11. 对话框\_通用消息对话框

12. 对话框\_文件对话框

13. 对话框\_颜色选择对话框

14. 几何管理器\_网络管理器

15. 几何管理器\_包管理器

16. 几何管理器\_位置管理器

17. 键盘事件处理

18. 鼠标事件处理

19. 动画的实现

20. 【项目】\_房贷计算器\_1

21. 【项目】\_房贷计算器\_2

22. 【项目】\_房贷计算器\_3

23. 【项目】\_房贷计算器\_4

24. 【项目】\_房贷计算器\_4

# 第12章 【项目】坦克大战游戏开发

1. pygame使用介绍\_简单游戏编程
2. pygame事件处理\_图形图像处理
3. pygame精灵\_碰撞检测

[4. 坦克大战面向对象分析](https://www.bz6000.cn/index/course/index/id/5528.html" \o " 坦克大战面向对象分析" \t "https://www.bz6000.cn/stages/id/_blank)

[5. 坦克大战之加载主页面](https://www.bz6000.cn/index/course/index/id/5529.html" \o " 坦克大战之加载主页面" \t "https://www.bz6000.cn/stages/id/_blank)

[6. 坦克大战之-添加左上角提示文字](https://www.bz6000.cn/index/course/index/id/5530.html" \o "坦克大战之-添加左上角提示文字" \t "https://www.bz6000.cn/stages/id/_blank)

[7. 坦克大战之添加事件监听](https://www.bz6000.cn/index/course/index/id/5531.html" \o "坦克大战之添加事件监听 " \t "https://www.bz6000.cn/stages/id/_blank)

[8. 坦克大战之加载我方坦克\_](https://www.bz6000.cn/index/course/index/id/5532.html" \o "坦克大战之加载我方坦克_" \t "https://www.bz6000.cn/stages/id/_blank)

[9. 坦克大战之实现我方坦克的移动](https://www.bz6000.cn/index/course/index/id/5533.html" \o "坦克大战之实现我方坦克的移动" \t "https://www.bz6000.cn/stages/id/_blank)

[10. 坦克大战之优化坦克移动方式](https://www.bz6000.cn/index/course/index/id/5534.html" \o "坦克大战之优化坦克移动方式" \t "https://www.bz6000.cn/stages/id/_blank)

[11. 坦克大战之新增随机生成敌方坦克](https://www.bz6000.cn/index/course/index/id/5535.html" \o " 坦克大战之新增随机生成敌方坦克" \t "https://www.bz6000.cn/stages/id/_blank)

[12. 坦克大战之实现敌方坦克的随机移动](https://www.bz6000.cn/index/course/index/id/5536.html" \o "坦克大战之实现敌方坦克的随机移动" \t "https://www.bz6000.cn/stages/id/_blank)

[13. 坦克大战之我方坦克发射子弹](https://www.bz6000.cn/index/course/index/id/5537.html" \o "坦克大战之我方坦克发射子弹" \t "https://www.bz6000.cn/stages/id/_blank)

[14. 坦克大战之子弹的优化处理](https://www.bz6000.cn/index/course/index/id/5538.html" \o "坦克大战之子弹的优化处理" \t "https://www.bz6000.cn/stages/id/_blank)

[15. 新增敌方坦克随机发射子弹？](https://www.bz6000.cn/index/course/index/id/5539.html" \o "新增敌方坦克随机发射子弹？" \t "https://www.bz6000.cn/stages/id/_blank)

[16. 敌方子弹与我方法坦克的碰撞以及我方坦克重生](https://www.bz6000.cn/index/course/index/id/5540.html" \o "敌方子弹与我方法坦克的碰撞以及我方坦克重生" \t "https://www.bz6000.cn/stages/id/_blank)

[17. 坦克大战之敌方坦克的碰撞检测](https://www.bz6000.cn/index/course/index/id/5541.html" \o " 坦克大战之敌方坦克的碰撞检测" \t "https://www.bz6000.cn/stages/id/_blank)

[18. 坦克大战之爆炸效果的实现](https://www.bz6000.cn/index/course/index/id/5542.html" \o "坦克大战之爆炸效果的实现" \t "https://www.bz6000.cn/stages/id/_blank)

[19. 爆炸效果补充](https://www.bz6000.cn/index/course/index/id/5543.html" \o "爆炸效果补充" \t "https://www.bz6000.cn/stages/id/_blank)

[20 坦克大战之音效处理](https://www.bz6000.cn/index/course/index/id/5544.html" \o "坦克大战之音效处理" \t "https://www.bz6000.cn/stages/id/_blank)

# 第13章 图形绘制

1. 基于turtle的画笔绘图\_矩形\_多边形\_圆形
2. 基于turtle的画笔绘图\_递归图形
3. 基于turtle的画笔绘图\_实操练习
4. 基于tkinter图形绘制\_矩形\_线条\_字符串
5. 基于tkinter图形绘制\_绘制函数图形
6. Matplotlib\_安装
7. Matplotlib\_简单图形绘制
8. Matplotlib\_函数曲线
9. Matplotlib\_多个图形
10. Matplotlib\_绘制直方图

# 第14章 数据库编程

1. 内置数据库SQLite

2. 操作SQLite数据库\_Connection\_Cursor\_Row

2. Mysql数据库的安装和配置

3. Mysql可视化客户端软件安装

4. PyMysql的使用

5. 对于数据库表的增删改查操作1

6. 对于数据库表的增删改查操作2

7. 对于数据库表的增删改查操作3

# 第15章 多线程和并发编程

[1：简述多进程](https://www.bz6000.cn/index/course/index/id/5545.html" \o "简述多进程" \t "https://www.bz6000.cn/stages/id/_blank)

[2：进程类](https://www.bz6000.cn/index/course/index/id/5546.html" \o "进程类" \t "https://www.bz6000.cn/stages/id/_blank)

[3：多进程默认不共享数据](https://www.bz6000.cn/index/course/index/id/5547.html" \o "多进程默认不共享数据" \t "https://www.bz6000.cn/stages/id/_blank)

[4：子类创建](https://www.bz6000.cn/index/course/index/id/5548.html" \o "子类创建" \t "https://www.bz6000.cn/stages/id/_blank)

[5：进程池](https://www.bz6000.cn/index/course/index/id/5549.html" \o "进程池" \t "https://www.bz6000.cn/stages/id/_blank)

[6：进程间通信1](https://www.bz6000.cn/index/course/index/id/5550.html" \o "进程间通信1" \t "https://www.bz6000.cn/stages/id/_blank)

[7：进程间通信2](https://www.bz6000.cn/index/course/index/id/5551.html" \o "进程间通信2" \t "https://www.bz6000.cn/stages/id/_blank)

[8：简述多线程](https://www.bz6000.cn/index/course/index/id/5552.html" \o "简述多线程" \t "https://www.bz6000.cn/stages/id/_blank)

[9：创建子线程](https://www.bz6000.cn/index/course/index/id/5553.html" \o "创建子线程" \t "https://www.bz6000.cn/stages/id/_blank)

[10：运行多线程程序](https://www.bz6000.cn/index/course/index/id/5554.html" \o "运行多线程程序" \t "https://www.bz6000.cn/stages/id/_blank)

[11：线程同步](https://www.bz6000.cn/index/course/index/id/5555.html" \o "线程同步" \t "https://www.bz6000.cn/stages/id/_blank)

[12：同步异步](https://www.bz6000.cn/index/course/index/id/5556.html" \o "同步异步" \t "https://www.bz6000.cn/stages/id/_blank)

[13：生产者消费者模式](https://www.bz6000.cn/index/course/index/id/5557.html" \o "生产者消费者模式" \t "https://www.bz6000.cn/stages/id/_blank)

[14：threadlocal变量](https://www.bz6000.cn/index/course/index/id/5558.html" \o "threadlocal变量" \t "https://www.bz6000.cn/stages/id/_blank)

# 第16章 网络编程

[1：网络基础](https://www.bz6000.cn/index/course/index/id/5560.html" \o "网络基础" \t "https://www.bz6000.cn/stages/id/_blank)

[2：端口号的作用](https://www.bz6000.cn/index/course/index/id/5561.html" \o "端口号的作用" \t "https://www.bz6000.cn/stages/id/_blank)

[3：协议](https://www.bz6000.cn/index/course/index/id/5562.html" \o "协议" \t "https://www.bz6000.cn/stages/id/_blank)

[4：socket编程](https://www.bz6000.cn/index/course/index/id/5563.html" \o "socket编程" \t "https://www.bz6000.cn/stages/id/_blank)

[5：使用udp协议发送数据](https://www.bz6000.cn/index/course/index/id/5564.html" \o "使用udp协议发送数据" \t "https://www.bz6000.cn/stages/id/_blank)

[6：使用udp协议接收数据](https://www.bz6000.cn/index/course/index/id/5565.html" \o "使用udp协议接收数据" \t "https://www.bz6000.cn/stages/id/_blank)

[7：tftp协议](https://www.bz6000.cn/index/course/index/id/5566.html" \o "tftp协议" \t "https://www.bz6000.cn/stages/id/_blank)

[8：tftp下载原理](https://www.bz6000.cn/index/course/index/id/5567.html" \o "tftp下载原理" \t "https://www.bz6000.cn/stages/id/_blank)

[9：tftp下载器](https://www.bz6000.cn/index/course/index/id/5568.html" \o "tftp下载器" \t "https://www.bz6000.cn/stages/id/_blank)

[10：udp广播](https://www.bz6000.cn/index/course/index/id/5569.html" \o "udp广播" \t "https://www.bz6000.cn/stages/id/_blank)

[11：tcp通信过程](https://www.bz6000.cn/index/course/index/id/5570.html" \o "tcp通信过程" \t "https://www.bz6000.cn/stages/id/_blank)

[12：tcp服务器](https://www.bz6000.cn/index/course/index/id/5571.html" \o "tcp服务器" \t "https://www.bz6000.cn/stages/id/_blank)

[13：多进程服务器](https://www.bz6000.cn/index/course/index/id/5572.html" \o "多进程服务器" \t "https://www.bz6000.cn/stages/id/_blank)

14. 聊天室项目\_1

15\_聊天室项目\_2

16\_聊天室项目\_3

1. 聊天室项目\_4
2. 聊天室项目\_5

# 第17章 协程和异步IO

1. 协程的概念
2. 定义一个协程
3. 创建task执行协程\_绑定回调
4. future 与 result
5. 阻塞和await
6. 并发和并行
7. 协程嵌套
8. 协程停止
9. master-worker主从模式
10. aiohttp

# 第18章 脚本化系统管理

1. 增强的交互式环境IPython\_IPython安装\_配置\_启动
2. 和IPython简单交互
3. magic函数
4. 执行外部系统命令和运行文件
5. 文件管理
6. 定时执行任务

# 第19章 语音识别模块\_中英文实时翻译项目

1. 语音识别基本原理
2. 文本转换语音
3. 调用讯飞动态链接库实现语音识别
4. 百度语音API调用
5. 【项目】实现中英文实时口语翻译\_1
6. 【项目】实现中英文实时口语翻译\_2
7. 【项目】实现中英文实时口语翻译\_3
8. 【项目】实现中英文实时口语翻译\_4
9. 【项目】实现中英文实时口语翻译\_5
10. 【项目】实现中英文实时口语翻译\_6

# 第20章 Windows下的Python开发

1. Python3安装pywin32模块

2. Windows编程基本知识\_组件对象

3. 实现COM组件

4. 对word的自动访问

5. 对excel的自动访问

6. 操作窗口

7. 操作进程

8. 操作线程

9. 桌面便签软件开发\_1

10. 桌面便签软件开发\_2

11. 桌面便签软件开发\_3

12. 桌面便签软件开发\_4

13. 桌面便签软件开发\_5

# 第21章 【实战】算法

1. 算法概述
2. 时间复杂度分析
3. 增长量级的概念
4. 空间复杂度分析
5. 查找算法\_顺序查找法
6. 查找算法\_二分查找法
7. 排序算法\_冒泡排序法
8. 排序算法\_选择排序法
9. 排序算法\_插入排序法
10. 排序算法\_归并排序法
11. 排序算法\_快速排序法
12. Python提供的查找算法
13. Python提供的排序算法

# 第22章 【实战】手写数据结构

1. 自定义队列\_队列特点\_入队

2. 自定义队列\_出队\_测试

3. 自定义栈\_入栈

4. 自定义栈\_出栈

5. 自定义二叉树\_分析和创建类

6. 自定义二叉树\_创建左子树\_创建右子树

7. 自定义二叉树\_前序遍历\_后序遍历\_中序遍历

8. 自定义二叉树\_测试

9. 自定义有向图\_特点分析

10.自定义有向图\_创建

11. 自定义有向图\_路径搜索功能

12. 自定义集合\_类的创建和分析

13. 自定义集合\_添加元素\_删除元素

14. 自定义集合\_差集运算

15. 自定义集合\_交集运算

16. 自定义集合\_并集运算

17. 自定义集合\_运算符重载\_集合包含关系处理

18. 自定义集合\_运算符重载\_支持in运算符

19. 手写数据结构总结

# 第23章 正则表达式和复杂文本操作

[1：正则表达式介绍\_特殊符号和字符](https://www.bz6000.cn/index/course/index/id/2651.html" \o "正则表达式介绍_特殊符号和字符" \t "https://www.bz6000.cn/stages/id/_blank)

[2：字符匹配](https://www.bz6000.cn/index/course/index/id/2652.html" \o "字符匹配" \t "https://www.bz6000.cn/stages/id/_blank)

[3：创建字符集\_表示字符集的特殊字符](https://www.bz6000.cn/index/course/index/id/2653.html" \o "创建字符集_表示字符集的特殊字符" \t "https://www.bz6000.cn/stages/id/_blank)

[4：限定范围和否定](https://www.bz6000.cn/index/course/index/id/2654.html" \o "限定范围和否定" \t "https://www.bz6000.cn/stages/id/_blank)

[5：使用小括号指定分组](https://www.bz6000.cn/index/course/index/id/2655.html" \o "使用小括号指定分组" \t "https://www.bz6000.cn/stages/id/_blank)

[6：贪婪模式和非贪婪模式](https://www.bz6000.cn/index/course/index/id/2656.html" \o "贪婪模式和非贪婪模式" \t "https://www.bz6000.cn/stages/id/_blank)

[7：常用正则表达式示例](https://www.bz6000.cn/index/course/index/id/2657.html" \o "常用正则表达式示例" \t "https://www.bz6000.cn/stages/id/_blank)

[8：re 模块\_核心函数和方法\_compile()方法](https://www.bz6000.cn/index/course/index/id/2658.html" \o "re 模块_核心函数和方法_compile()方法" \t "https://www.bz6000.cn/stages/id/_blank)

[9：匹配对象和 group()以及 groups()方法](https://www.bz6000.cn/index/course/index/id/2659.html" \o "匹配对象和 group()以及 groups()方法" \t "https://www.bz6000.cn/stages/id/_blank)

[10：match()匹配字符串](https://www.bz6000.cn/index/course/index/id/2660.html" \o "match()匹配字符串" \t "https://www.bz6000.cn/stages/id/_blank)

[11：search()在一个字符串中查找](https://www.bz6000.cn/index/course/index/id/2661.html" \o "search()在一个字符串中查找" \t "https://www.bz6000.cn/stages/id/_blank)

[12：案例：编程实现抓取 163 主页所有图片地址信息](https://www.bz6000.cn/index/course/index/id/2662.html" \o "案例：编程实现抓取 163 主页所有图片地址信息" \t "https://www.bz6000.cn/stages/id/_blank)

# 第24章 函数式编程和高阶函数

1. 函数式编程是什么

2. 高阶函数是什么

3. 将函数作为参数

4. 返回值为函数

5. map()高阶函数\_1

6. map()高阶函数\_2

7. map()高阶函数\_3

8. reduce()高阶函数\_1

9. reduce()高阶函数\_2

10. reduce()高阶函数\_3

11. filter()高阶函数

12. sorted()高阶函数

13. 闭包\_1

14. 闭包\_2

15. 闭包\_3

16. 匿名函数

17. 装饰器\_1

18. 装饰器\_2

19. 装饰器\_3

20. 偏函数

# 第25章 测试

1. 单元测试\_unittest

2. 单元测试\_coverage

3. 性能测试\_timeit

4. 性能测试\_profile

5. 性能测试\_line profiler

6. 性能测试\_memory profiler

7. 性能测试\_pympler

# 第26章 【项目】手工实现神经网络

1. 神经网络基本原理\_1
2. 神经网络基本原理\_2

3. MNIST数据集简介

4. 矩阵原理\_1

5. 矩阵原理\_相关运算\_2

6. 正态分布原理\_1

7. 正态分布原理\_2

8. Sigmoid函数\_神经网络激活函数\_1

9. Sigmoid函数\_神经网络激活函数\_2

10. 使用matplotlib绘制Sigmoid函数

11. 前向反馈\_feedforward函数\_1

12. 前向反馈\_feedforward函数\_2

13. 前向反馈\_feedforward函数\_3

14. 神经网络学习能力构建\_随机梯度下降函数\_1

15. 神经网络学习能力构建\_随机梯度下降函数\_2

16.神经网络学习能力构建\_随机梯度下降函数\_3

17.神经网络学习能力构建\_随机梯度下降函数\_4

18.神经网络学习能力构建\_随机梯度下降函数\_5

19.MNIST数据集的加载

20.测试和训练神经网络