



北京航空航天大学
BEIHANG UNIVERSITY

大学计算机基础

北京航空航天大学
计算机学院



文件

- 计算的输入数据和结果通常使用文件来存储
 - ◆ Python提供多种便利进行文件创建及访问
- 每种操作系统有自己相应的文件系统以进行文件的创建和访问
 - ◆ Python通过文件句柄(file handle)访问文件实现操作系统无关性
 - ◆ `nameHandle = open('kids.txt', 'w')`
 - ✓ 通知操作系统创建一个名为kids.txt的文件，并返回该文件的文件句柄
 - ✓ 参数'w'指明文件可写





文件操作

■ 扩展：文件操作

为了在程序执行完后仍保存结果，可用文件实现。

- ◆ 打开：fileobject = open('/路径/文件名.后缀名')
- ◆ 关闭：fileobject.close()
- ◆ 读取：fileobject.read()或fileobject.readline()或fileobject.readlines()
- ◆ 写入：fileobject = open('/路径/文件名.后缀名', 'w')
fileobject.write()或fileobject.writelines()

注意：(1)文件在打开时默认是只读模式，如果想写入必须修改为可写模式，修改模式在open函数中实现。

(2)文件读取时有三种方法：read(), readline()和readlines()

(3)文件写入时两种方法：write()或writelines()





文件操作

r	以读方式打开文件，可读取文件信息。
w	以写方式打开文件，可向文件写入信息。如文件存在，则清空该文件，再写入新内容
a	以追加模式打开文件（即一打开文件，文件指针自动移到文件末尾），如果文件不存在则创建
r+	以读写方式打开文件，可对文件进行读和写操作。
w+	消除文件内容，然后以读写方式打开文件。
a+	以读写方式打开文件，并把文件指针移到文件尾。
b	以二进制模式打开文件，而不是以文本模式。该模式只对Windows或Dos有效，类Unix的文件是用二进制模式进行操作的。

read()	每次读取整个文件，它通常用于将文件内容放到一个字符串变量中
readline()	每次读取一行文件，返回一行字符串
readlines()	每次读取整个文件，返回包含文件所有内容的字符串列表，每个元素是一行的字符串
write()	写入内容后光标在行末不会换行，下次写会接着这行写
writelines()	将输入列表写入文件



写文件时如需换行，需要写入'\n'





示例

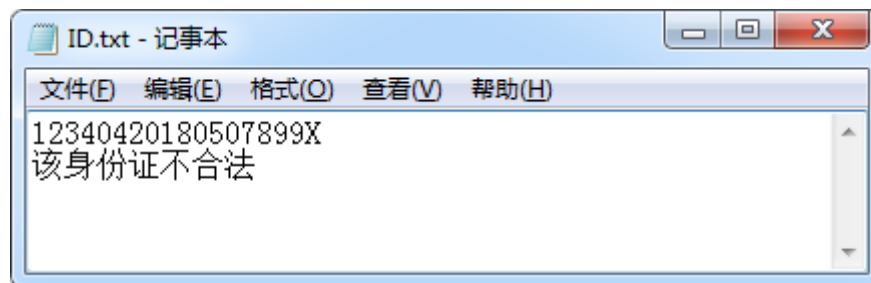
- (1) 打开一个文件并写入
 - ◆ `f = open('hello.txt','w')`
 - ✓ 注意：如果文件不存在，就会新建建立一个文件，
 - ◆ `f.write('hello my friend python!')`
 - ✓ 注意：会清空文件写入新内容
 - ✓ 注意：该语句有返回值，返回的是写入的字符总数
 - ◆ `f.close()`
- (2) 打开一个文件，读取其中的内容
 - ◆ `f = open('hello.txt','r')`
 - ◆ `content=f.read()`
 - ◆ `f.close()`
 - ◆ `print(content)`





课内练习1

- 请将4月9日提交的身份证验真课内作业代码进行修改，实现从txt文件中读取待核验的身份证号，并将结果输出到该txt文件末尾（单起一行）



- 注意：readline()读入的末尾字符是什么？
- 注意：单起一行需要先写入一个'\n'

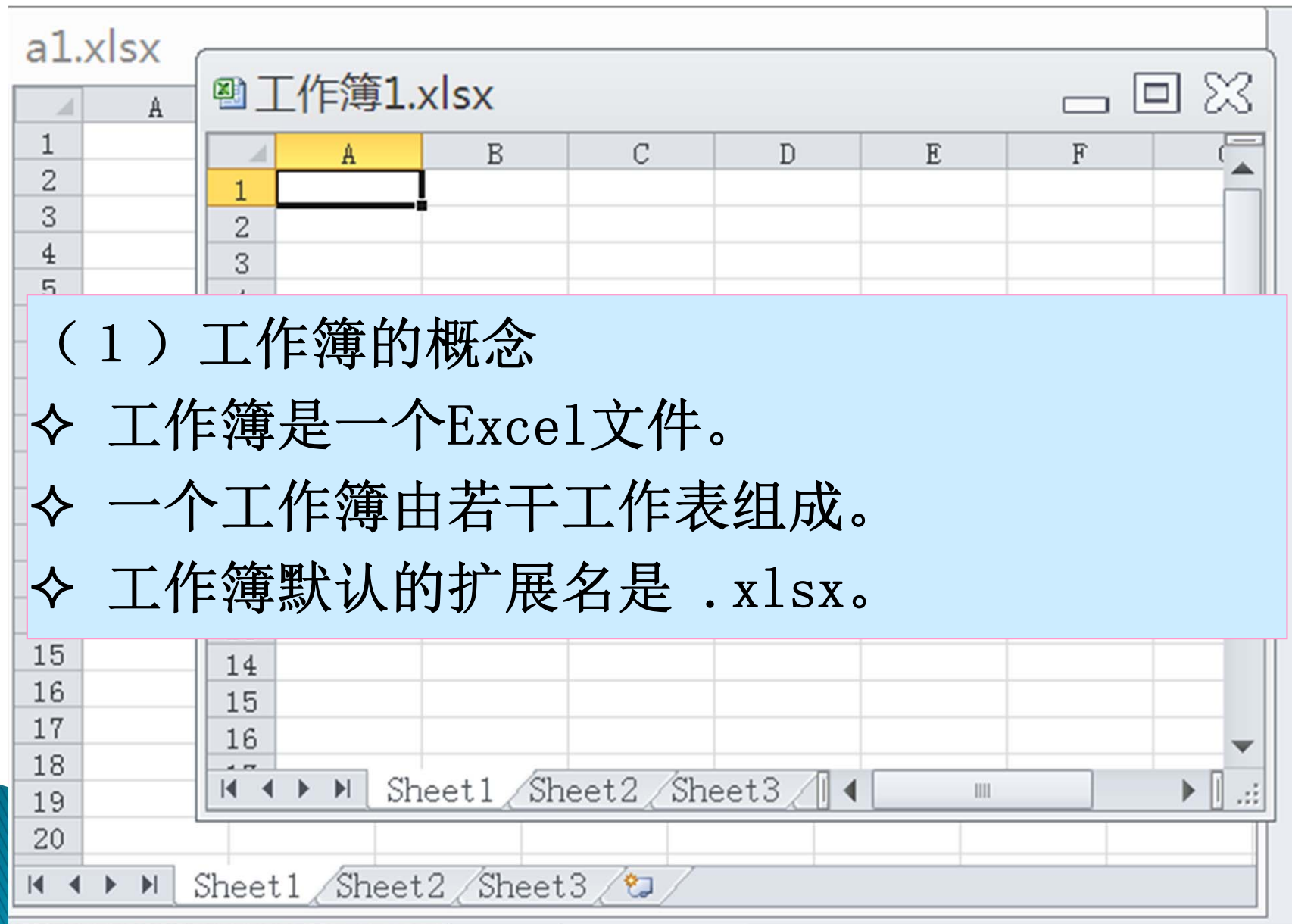


Excel简介

■ Excel电子表格软件的特点：

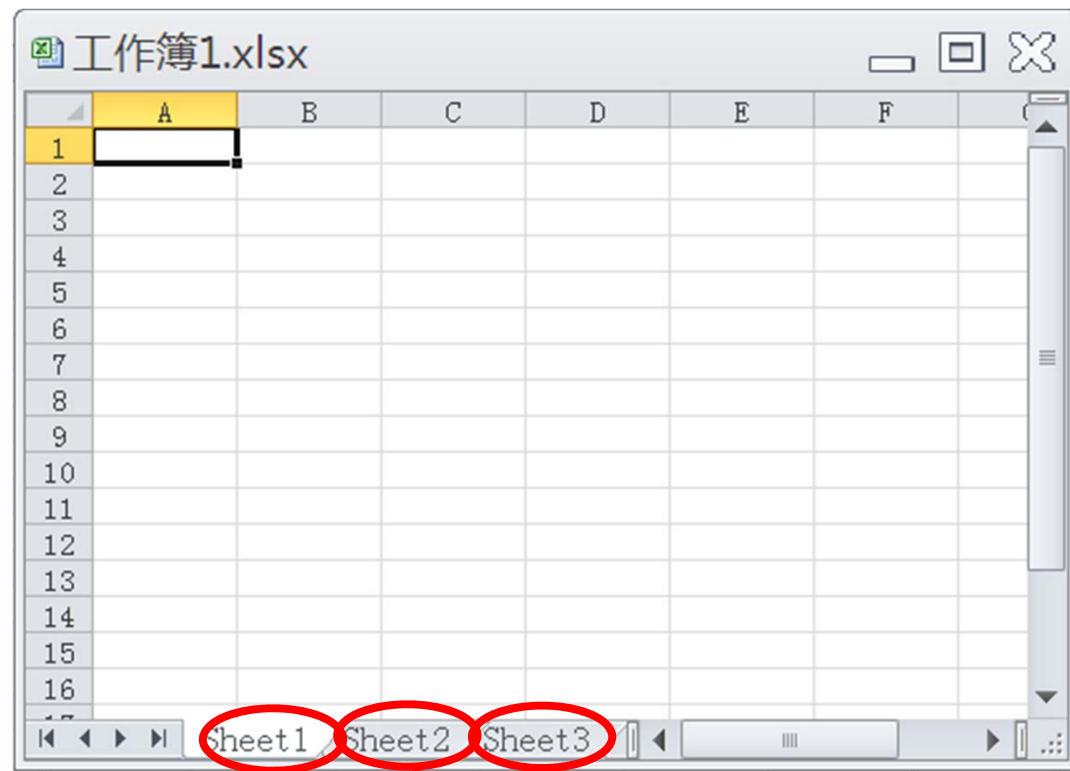
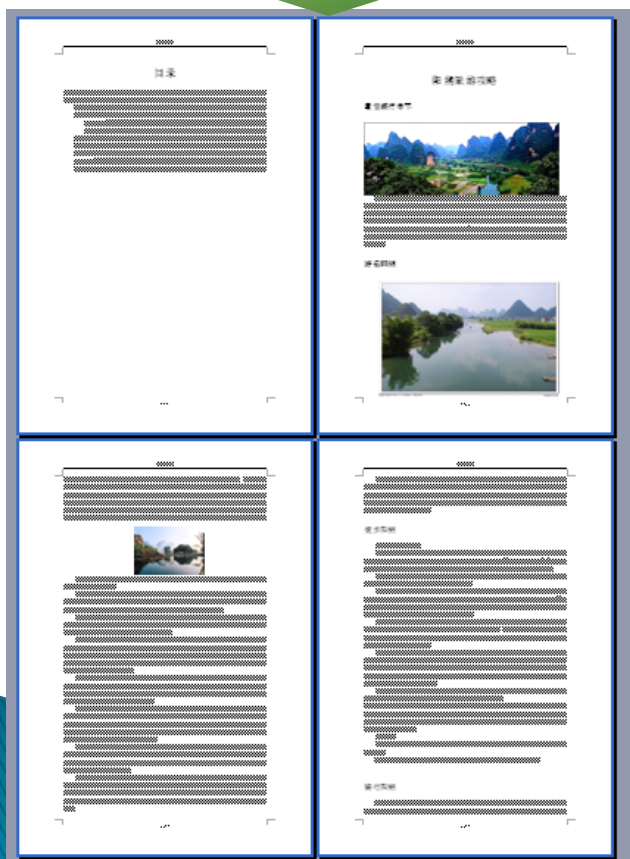
- ① Excel 是一个功能强大的电子表格软件。
- ② 具有强大的数据输入、编辑、计算与分析处理功能。
- ③ 将数据用表格及各种统计图、透视图的形式表示，使制作出的报表图文并茂，信息表述更清晰。

工作簿



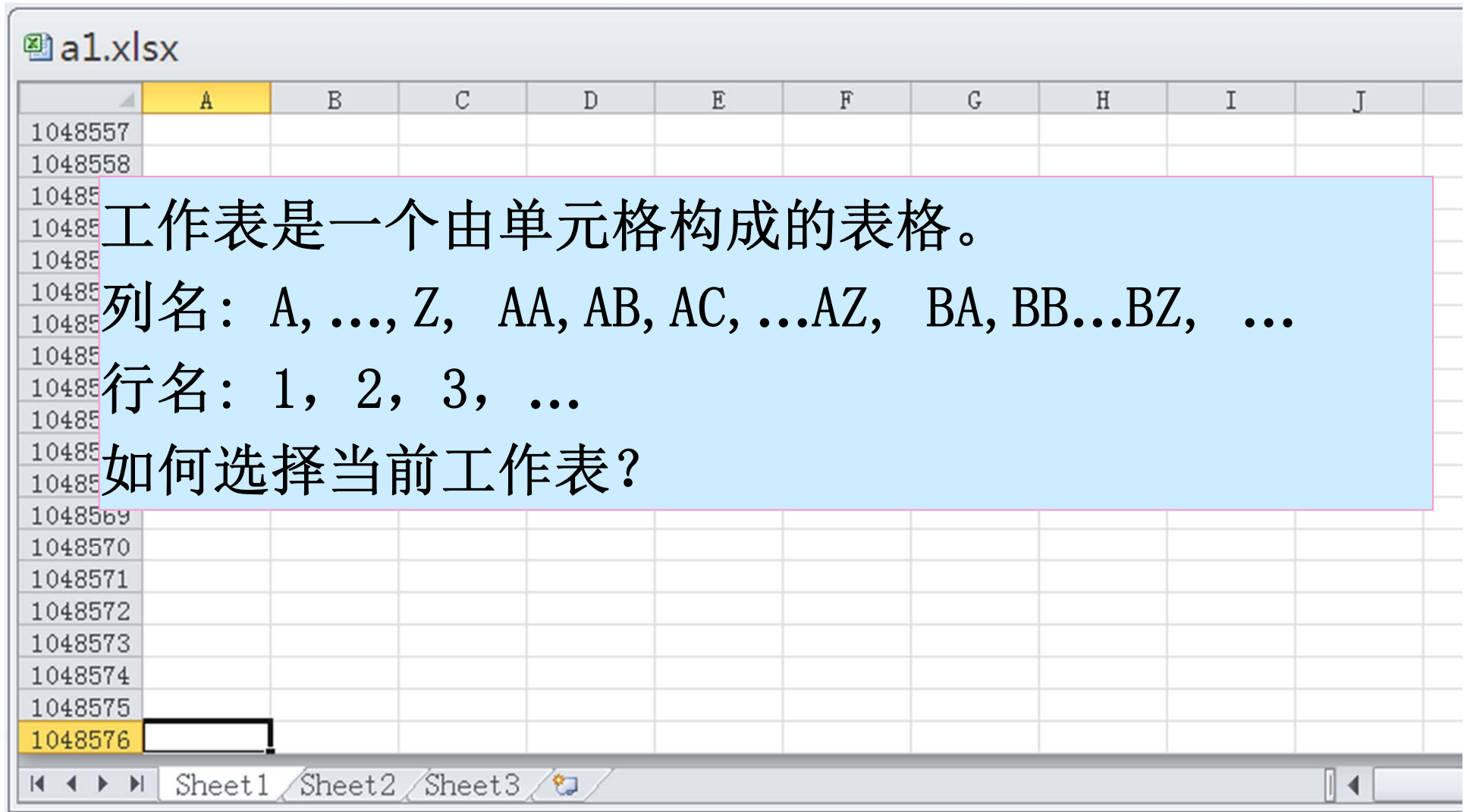
Excel文档的组成——工作表

Word中有若干页

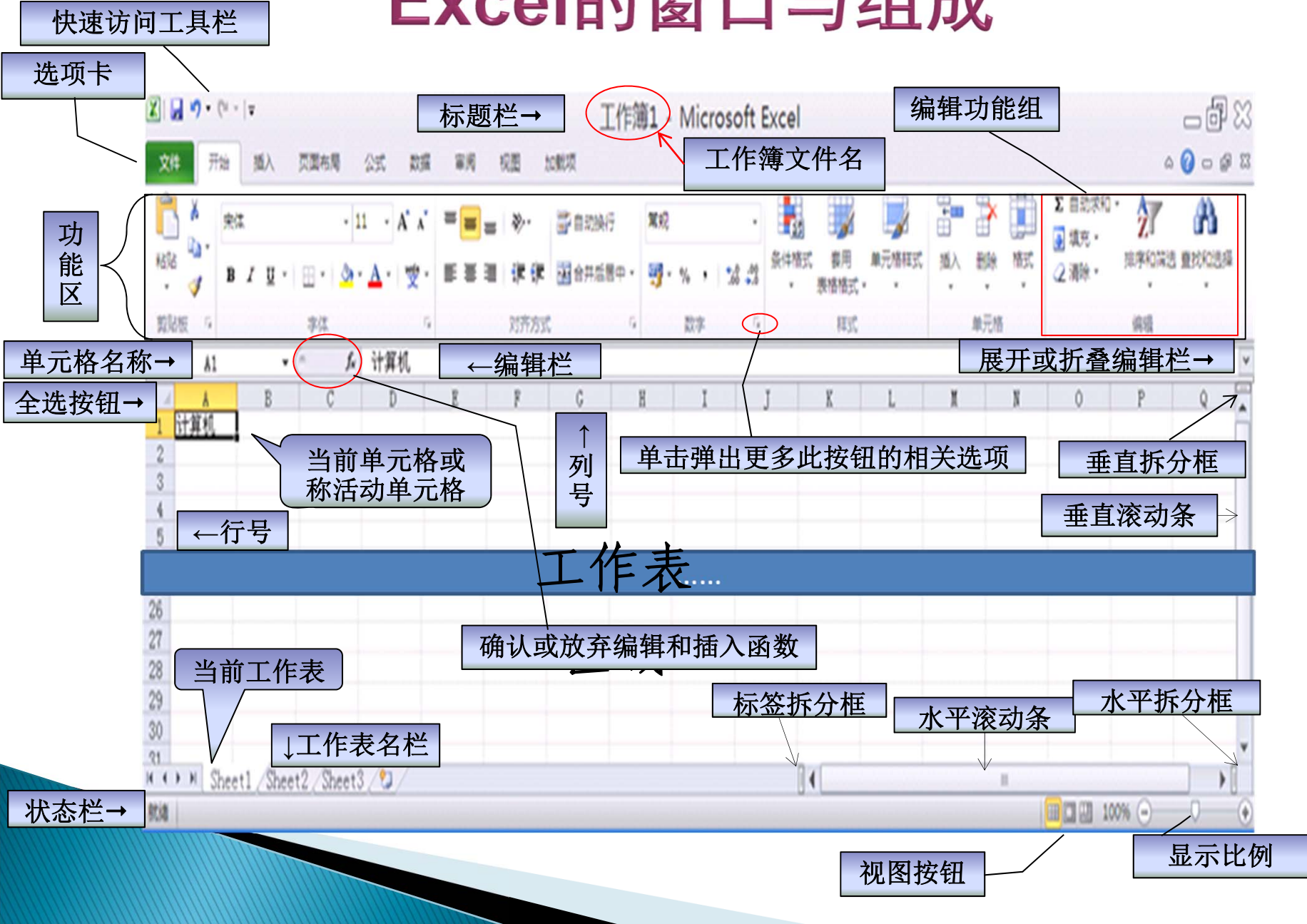


Excel中有若干“工作表”

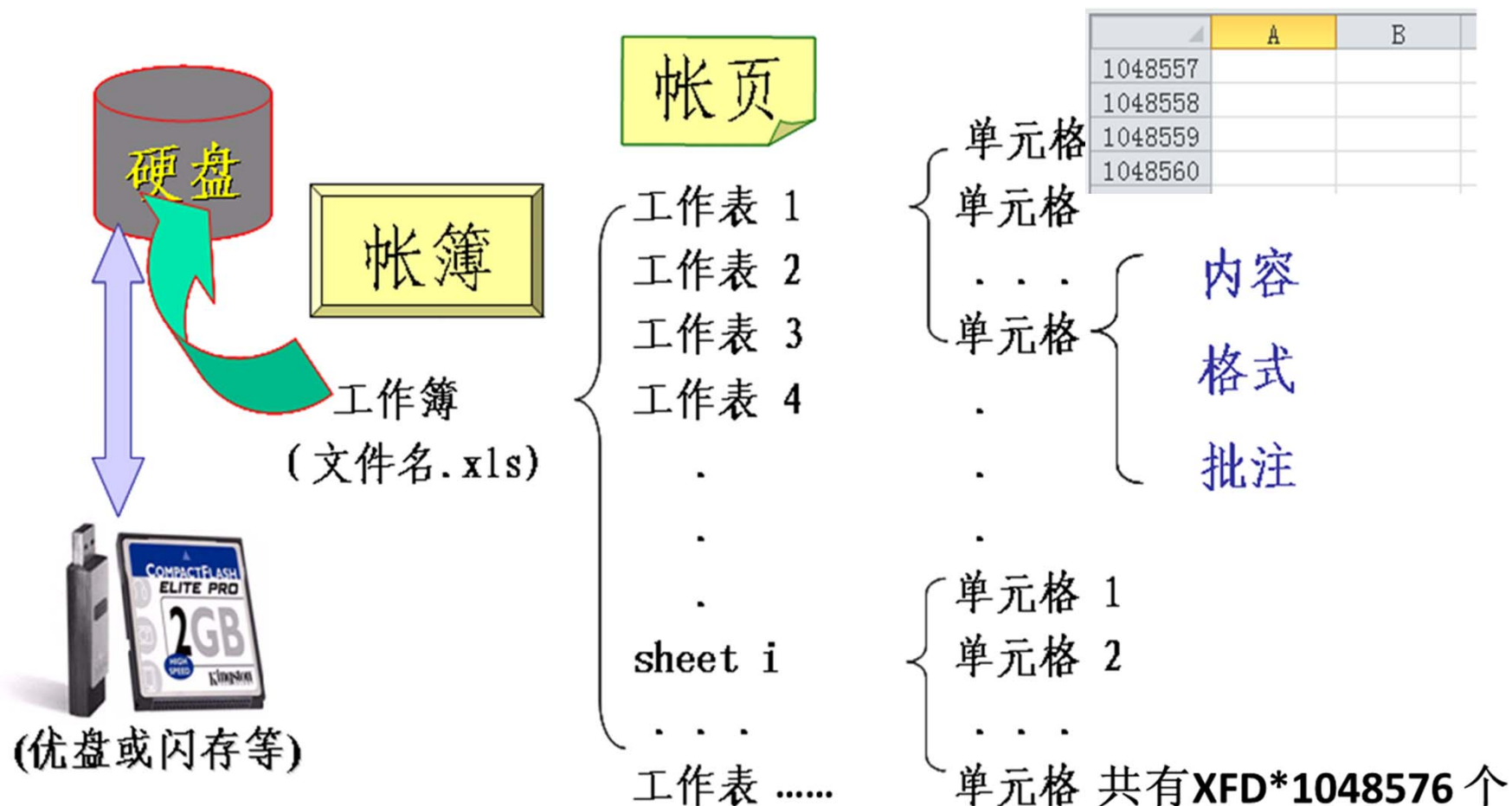
工作表



Excel的窗口与组成



工作簿、工作表和单元格的关系



单元格

shanghai

单元格是工作表中最基本的存储数据单元。

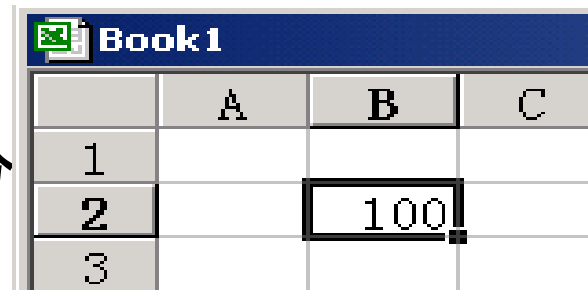
单元格的命名是由列标和行号来标识。例：

B2表示第**2**列第**2**行的单元格

A18表示第1列第18行的单元格

单元格的数据信息

- ▶ 一个**单元格**的信息可以包含三部分



	A	B	C
1			
2		100	
3			

内容——文字、数字、日期、时间、公式(包括函数)。

批注——帮助理解该单元格数据的含义，或者作为数据的说明。

格式——系统提供缺省格式。也可以重新对单元格进行格式设置。（格式设置信息包括数据的字体、字体大小、颜色、边框及其线形等）。

区域

a1.xlsx

	A	B	C	D	E
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					

区域的表示方法
左上角单元格名:右下角单元格名
例: b2:d5 ≡ B2:D5

单元格、区域的选择

- ▶ 1. 用鼠标选定。
- ▶ 2. 用鼠标和键盘（Shift键、Ctrl键等）相结合选定。
- ▶ 3. 定位
- ▶ 4. 撤消选择——只要重新选择其它单元格。

怎样初步保证数据的正确？

- ▶ 选择要定义有效数据的单元格或单元格区域
 - 数据|数据有效性

E	F	G	H
高等数学	外语	计算机	物理
87	90	95	70
95	95	70	70
70	95	75	90
70	66	95	70
90	85	95	53
70	90	95	66
91	70	-13	85
90	94	90	90
95	70	55	70
59	67	90	94
87	70	85	70
90	57	78	90
85	97	95	85
85	105	85	78
95	89	100	48
20	70	56	71
85	85	70	70



数据有效性

设置 输入信息 出错警告 输入法模式

有效性条件

允许(A): 整数 ☒ 忽略空值(B)

数据(D): 介于

最小值(M)

最大值(X)

☐ 对具有同样设置的所有其他单元格应用这些更改(P)

全部清除(C) 确定 取消

1) 使用公式计算学生的总成绩

	A	B	C	D	E	F	G	H	I
1	学号	姓名	性别	生日	高等数学	外语	计算机	物理	总成绩
2	37150101	秦博	男	1992-1-1	87	90	95	70	=
3	37150102	唐磊	女	1992-6-2	95	95	70	70	
4	37150103	高峰	男	1992-7-3	70	95	75	90	
5	37150104	王珠	女	1992-10-26	70	66	95	70	

以“=”开头

- ▶ 使用公式和函数都可以计算总成绩
- ▶ 复制公式

公式中的注意问题

- ▶ 公式的计算和嵌套
- ▶ 公式中的数值转换
- ▶ 日期和时间的运算

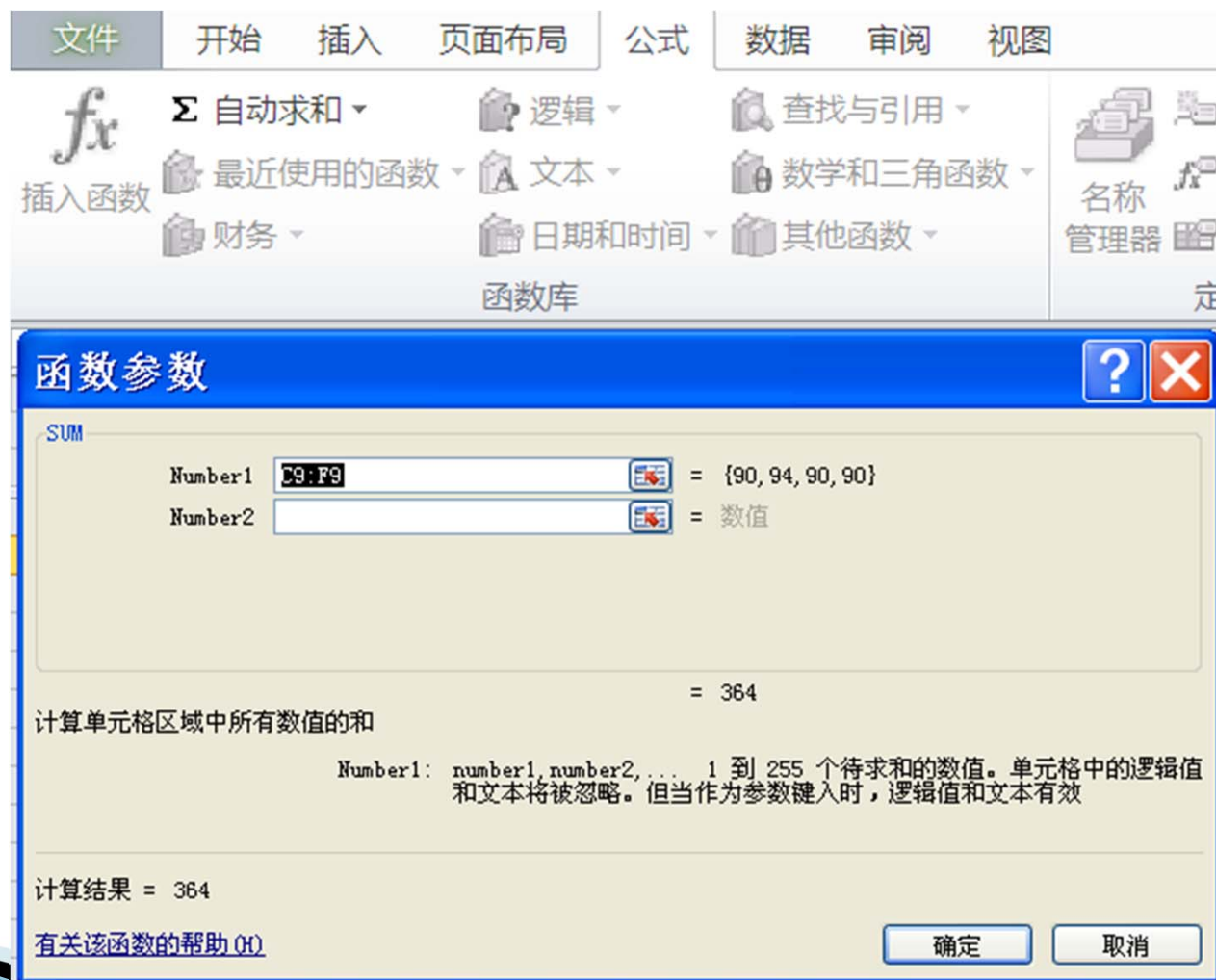
使用函数计算学生的总成绩

G9		fx =SUM(C9:F9)				
A	B	C	D	E	F	G
学号	姓名	高等数学	外语	计算机	物理	总成绩
37150101	秦博	87	90	95	97	369
37150107	黎波	91	70	-13	85	233
37150108	张健	90	94	90	90	364

从键盘上直接输入 “=sum(E2:H2)”

不知道函数名称怎么办?

学会插入函数的方法



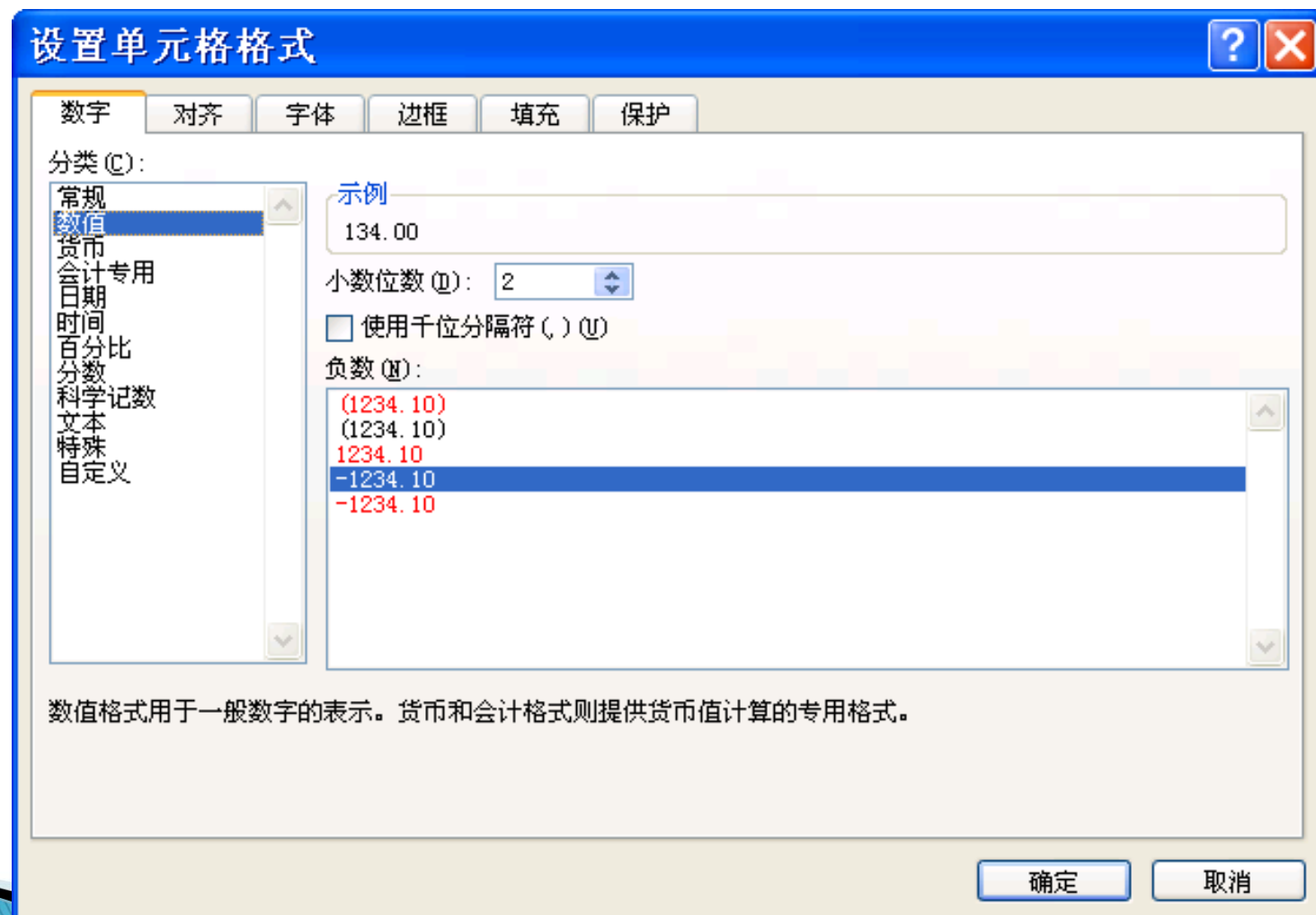
Excel的函数

- ▶ Excel的函数只能出现在公式中。
- Excel提供了大量的**内置函数**，为用户对数据运算和分析带来极大方便。范围包括：常用、财务、日期与时间、数学与三角函数、统计、查找与引用、数据库、文本、逻辑、信息等。
- 当选用某一函数后，在弹出的【函数参数】对话框中将会显示各个参数相应的提示信息。

公式中的运算符

- ▶ 在Excel公式中，‘=’后面是参与计算的运算数和运算符。
- **运算数**：可以是常量数值、单元格或单元格区域的引用、标志名称以及函数引用。
- **运算符**：有算术、比较、文本和引用运算符。
- **算术运算符**：**-**（单目负号）、**%**（百分比）、**^**（乘幂）、*****（乘）、**/**（除）、**+**、**-**。
- **比较运算符**：**=**、**<>**（不相等）、**>**、**>=**、**<**、**<=**（结果为**TRUE**或**FALSE**）。
- **文本运算符**：**&**（连接）。
- **引用运算符**：**:**（区域）、**,**（并集）、**空格**（交集）。
- **运算优先级**：**()** 括号先做，再考虑引用运算符（**:**、**,**、**空格**）、算术和比较运算符。同级运算从左到右。

Excel格式设置





Python 操作 Excel

- openpyxl是一个用来读写Excel文件的python模块。例如：
 - ◆ 你可能需要从一个表格按照一定的格式将数据复制到另外一个表格中
 - ◆ 你可能需要查找上千条数据中满足某个条件的几条，并且修改这些条目





Python 操作 Excel

- 实验室的机器上都已经安装了openpyxl，如果你要在自己的电脑上进行操作，那么你需要安装openpyxl包。我们推荐使用pip进行安装，运行命令：

```
pip install openpyxl
```

- 在安装好openpyxl之后，如果想要在我们的程序中使用openpyxl，还需要导入包。加入下列语句即可



```
import openpyxl
```





Python 操作 Excel

- 打开Excel文件，打开工作表
- 访问单元格
- 访问一整行或一整列
- 创建并保存Excel文件
- 创建与删除工作表
- 向单元格中写入值





Python 操作 Excel

■ 示例数据介绍

Iris 鸢尾花

自然界中，存在3种Iris，为了研究这种花的分类，科学家分别采集了50朵（共150朵），测量了花瓣长度、花瓣宽度、萼片长度和萼片宽度（cm），这个数据成为了研究者最为熟悉的数据集（Dataset）

我们将数据存放在一个Excel的两个Sheet中，80%（三种各40朵）命名为Training，20% 命名为Testing

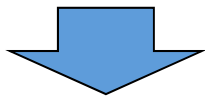




打开Excel文件与工作表

import openpyxl#导入包

```
#通过load_workbook打开excel
wb = openpyxl.load_workbook('iris.xlsx')
print(wb.sheetnames)
#wb是一个字典，获取一个Sheet对象
sheet = wb['Testing']
#查看Sheet的名字 -Testing
print(sheet.title)
```



```
===== RESTART: C:/Users/Jason/
['Training', 'Testing', 'Sheet3']
Testing
>>> |
```



1. openpyxl.open_workbook()
函数打开参数中的文件名并返回类型为Workbook的工作簿对象。这个工作簿对象代表对应的Excel文件

2. 要注意iris.xlsx必须和程序或者交互式命令行在同一个目录下

3. 每个工作表由一个Worksheet对象表示，你可以通过向Workbook对象传入一个工作表名称字符串来得到它

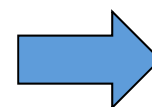




访问单元格

```
import openpyxl#导入包
```

```
#通过load_workbook打开excel
wb = openpyxl.load_workbook('iris.xlsx')
#wb是一个字典，获取一个Sheet对象
sheet = wb['Training']
#sheet又是一个字典，可以通过单元格的位置访问
print(sheet['A1'].value)
print(sheet['B1'].value)
#c是一个cell，代表单元格
c = sheet['A1']
#value是取值、取内容
print(c.value)
#row 获取本单元格所在行号
print(c.row)
#column获取列号
print(c.column)
```



=====
5.1
3.5
5.1
1
A
...

Cell对象有一个value属性，包含了单元格中的值。Cell对象还有row，column等属性

还可以写成：

sheet.cell(row=1, column=2).value





访问一整行或一整列

```
import openpyxl#导入包
```

```
#通过load_workbook打开excel
```

```
wb = openpyxl.load_workbook('iris.xlsx')
```

```
#wb是一个字典，获取一个Sheet对象
```

```
sheet = wb['Training']
```

```
#sheet支持切片，切片之后是一行
```

```
#对每一行进行循环，可以获取每一个cell
```

```
for rows in sheet['A1':'D2']:
```

```
    for c in rows:
```

```
        print(c.value)
```

=====

5.1

3.5

1.4

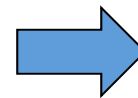
0.2

4.9

3

1.4

0.2





小结

- 导入openpyxl模块
- 调用openpyxl.load_workbook()函数
- 获取想要操作的Workbook对象
- 获取想要操作的Worksheet对象
- 获取想要操作的Cell对象
- 读取Cell对象的value属性





新建Excel与写入

```
import openpyxl#导入包

#Workbook直接创建一个Excel，它含有一个sheet
wb = openpyxl.Workbook()
#create_sheet 会创建sheet1
wb.create_sheet()
#create_sheet 指定参数，可以决定sheet创建的顺序和名字
wb.create_sheet(index=0, title='First Sheet')
sheet = wb['First Sheet']
#对cell直接赋值写入数据
sheet['A1'] = "hello"
#保存excel文件
wb.save('new.xlsx')
```





Excel 操作示例

- 分类器是机器学习中最常见的算法，其核心是通过“已知”解决“未知”
- 对Iris，如果我们已知3种花的各40个测量结果，对于一朵新采集的花，能否通过机器自动判定其所属类别呢？





简单分类器思路

- Training视为已知
- Testing视为未知
- 思路
 - ◆ 通过Training数据，对每种花的四个属性计算均值，获取这种花的“一般特性”，比如评价花瓣有多长？花瓣有多宽？ ...
 - ◆ 对于未知的一朵花，比较其与哪一类最为接近，则判定为那一类
 - ◆ 相似性判定方法：欧氏距离





课内练习2

- 请将4月16日提交的美国七个州和郡名称的课内作业中，描述州和郡数据的字典内容写入到excel文件，每个州名、缩写、郡名占用一个单元格

USState - 副本.xlsx - Excel

文件 开始 插入 页面布局 公式 数据 审阅 视图 告诉我 共享

J15

	A	B	C	D	E	F	G
1	name	abbreviation	counties				
2	Alabama	AL	Dale	Autauga	Baldwin	Bibb	Bullock
3	Alaska	AK	Valdez-Co	Wade Hamp	Wrangell	Yukon-Koy	Ketchikar
4	Arizona	AZ	Mohave	Santa Cru	Cochise	Graham	La Paz
5	Arkansas	AR	Cleveland	Columbia	Baxter	Cleburne	Cross
6	California	CA	Colusa	Glenn	Kings	Mariposa	Modoc
7	Colorado	CO	Hinsdale	Kiowa	Lincoln	Mineral	Montrose
8	Connectic	CT	Tolland	Windham	Fairfield	Hartford	Litchfield
9							

就绪





课内练习2

■ 提示：

- ◆ `SheetName.append(L)`：将列表L作为一行写入 excel文件，以第一行（为例）：

```
wb = openpyxl.Workbook()
sheet = wb['Sheet']
sheet.append(['name', 'abbreviation', 'counties'])
```

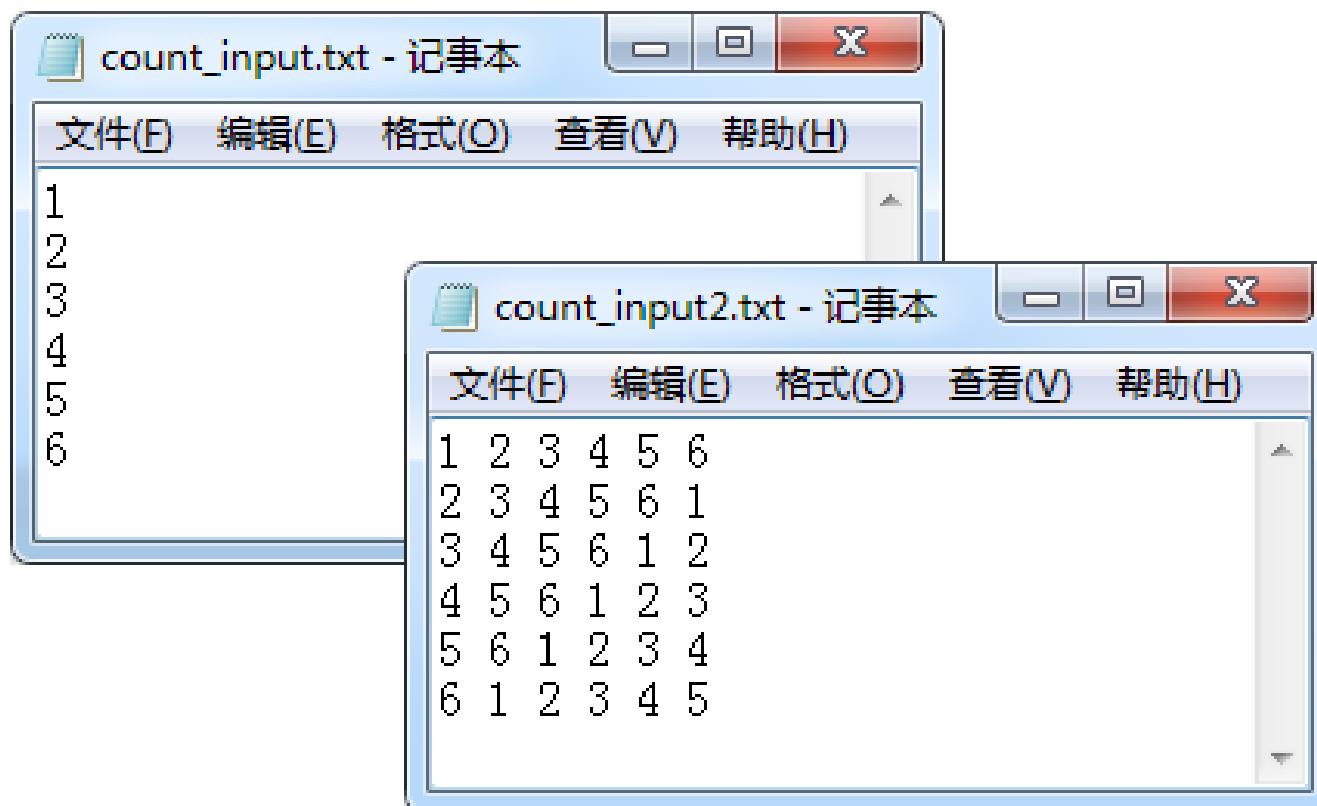
	A	B	C	D	E	F	G	
1	name	abbreviation	counties					
2	Alabama	AL	Dale	Autauga	Baldwin	Bibb	Bullock	C
3	Alaska	AK	Valdez-Co	Wade Ham	Wrangell	Yukon-Koy	Ketchikar	A
4	Arizona	AZ	Mohave	Santa Cru	Cochise	Graham	La Paz	P
5	Arkansas	AR	Cleveland	Columbia	Baxter	Cleburne	Cross	D
6	Californi	CA	Colusa	Glenn	Kings	Mariposa	Modoc	S
7	Colorado	CO	Hinsdale	Kiowa	Lincoln	Mineral	Montrose	P
8	Connectic	CT	Tolland	Windham	Fairfield	Hartford	Litchfiel	M
9								





热身题1

- 我们在4月16日课上曾经给出过计算平均分和均值的样例，请将该样例的输入方式修改为txt文件





热身题2

- 请尝试读取课内练习2生成的xlsx文件，将其内容转换为我们给出的[字典]形式，使用第一行单元格的内容作为字典的key

```
political_divisions = [{"name": "Alabama", "abbreviation": "AL",  
    "counties": ["Dale", "Autauga", "Baldwin", "Bibb", "Bullock", "Chilton", "Coffee", "Coosa",  
        "Etowah", "Franklin", "Lamar", "Macon", "Marion", "Morgan", "Randolph", "Sumter",  
        "Wilcox", "Barbour", "Blount", "Butler", "Calhoun", "Chambers", "Cherokee",  
        "Choctaw", "Clarke", "Clay", "Cleburne", "Colbert", "Conecuh", "Covington",  
        "Crenshaw", "Cullman", "Dallas", "DeKalb", "Elmore", "Escambia", "Fayette",  
        "Geneva", "Greene", "Hale", "Henry", "Houston", "Jefferson", "Lauderdale",  
        "Lawrence", "Lee", "Lowndes", "Madison", "Marengo", "Marshall", "Mobile",  
        "Monroe", "Montgomery", "Jackson", "Perry", "Pickens", "Pike", "Russell",  
        "St. Clair", "Shelby", "Talladega", "Tallapoosa", "Tuscaloosa", "Walker",  
        "Washington", "Winston", "Limestone"]},  
    {"name": "Alaska", "abbreviation": "AK",  
    "counties": ["Valdez-Cordova", "Wade Hampton", "Wrangell", "Yukon-Koyukuk", "Ketchikan Gateway",  
        "Aleutians East", "Bethel", "Hoonah-Angoon", "Sitka", "Kenai Peninsula",  
        "Kodiak Island", "Lake and Peninsula", "Nome", "Petersburg", "North Slope",  
        "Northwest Arctic", "Southeast Fairbanks", "Yakutat", "Aleutians West",  
        "Anchorage", "Bristol Bay", "Denali", "Dillingham", "Fairbanks North Star",  
        "Haines", "Juneau", "Matanuska-Susitna", "Prince of Wales-Hyder", "Skagway"]},
```

