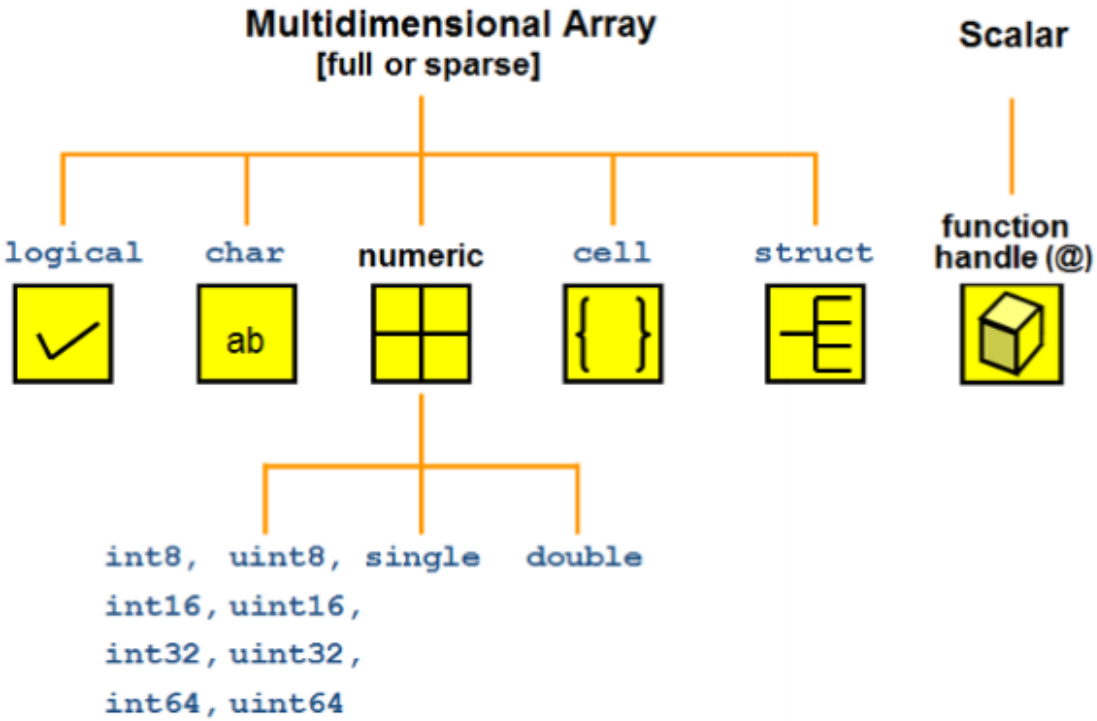
学习一门技术最好的方式就是阅读官方文档,可以查看MATLAB官方文档

数据类型

MATLAB中主要的数据类型如下:



下面依次介绍各种主要的数据类型,MATLAB官方文档介绍了所有数据类型.

数值类型(numeric)

在MATLAB中,数值类型的变量被默认为 `double` 类型的,可以使用类型转换将其转换为其他数值类型.

```
n = 3;
class(n)    % 得到 double

n = int8(3);
class(n)    % 得到 int8
```

MATLAB支持的数值类型见下表:

数值类型	描述
`double`	双精度浮点数
`single`	单精度浮点数
`int8`	8位带符号整数
`int16`	16位带符号整数
`int32`	32位带符号整数
`int64`	64位带符号整数
`uint8`	8位无符号整数
`uint16`	16位无符号整数
`uint32`	32位无符号整数
`uint64`	64位无符号整数

字符串类型(char)

- 在MATLAB中,字符串类型由一对单引号`'`包裹一段文字来定义.标准ASCII字符可以被转换为对应的ASCII码.

```
s1 = 'h';
uint16(s1)  % 得到 104
```

- 字符串在内存中是以字符矩阵的形式存储的,可以对其进行矩阵的索引以及赋值操作:

```
str1 = 'hello';
str2 = 'world';

str3 = [str1 str2];
size(str3)      % 得到 [1 10]

str4 = [str1; str2];
size(str4)      % 得到 [2 5]
```

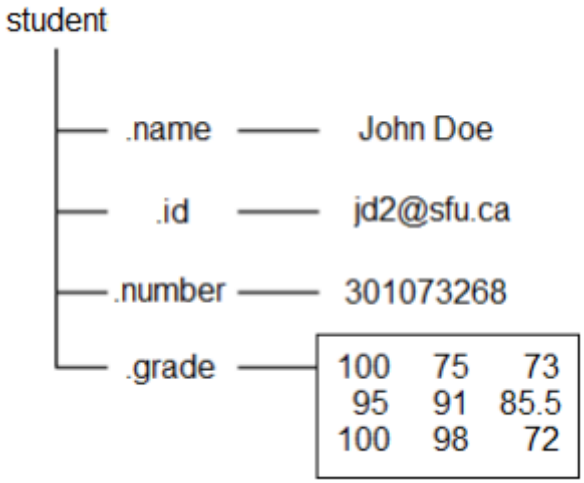
```
str = 'aardvark';
'a' == str              % 得到 [1 1 0 0 0 1 0 0]
str(str == 'a') = 'z'   % 得到 'zzrdvzrk'
```

结构体(structure)

在MATLAB中,结构体是一个存储 `{键: 值}` 的数据结构,类似于Python语言中的字典.

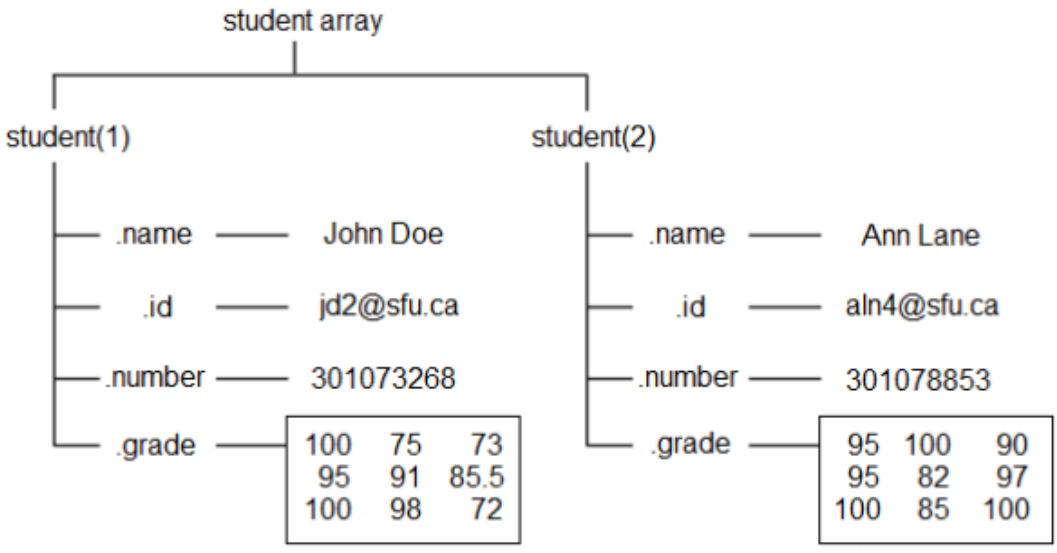
结构体的基本使用

- 与大多数编程语言类似,MATLAB使用`.`来访问结构体中的字段:



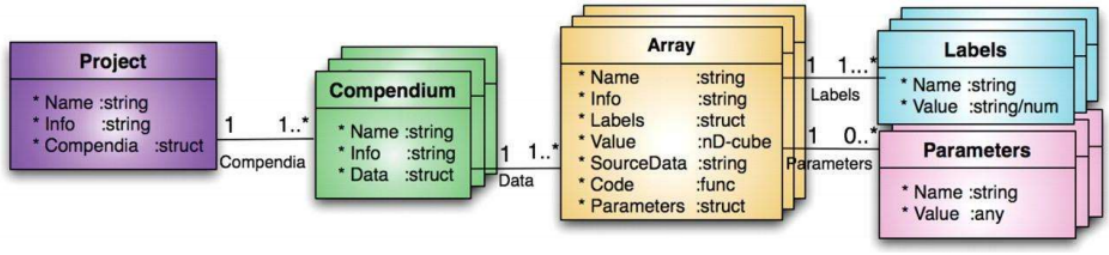

```
student.name = 'John Doe';
student.id = 'jdo2@sfu.ca';
student.number = 301073268;
student.grade = [100, 75, 73; ...
                 95, 91, 85.5; ...
                 100, 98, 72];
student
```

- 对结构体列表使用下标表达式可以扩充或缩减结构体列表.




```
student(2).name = 'Ann Lane';
student(2).id = 'aln4@sfu.ca';
student(2).number = 301078853;
student(2).grade = [95 100 90; 95 82 97; 100 85 100];
student

student(1) = []     % 删除student列表第一项
```

- 结构体可以级联,即结构体中字段的取值也可以是结构体:




```
A = struct('data', [3 4 7; 8 0 1], ...
    'nest', struct('testnum', 'Test 1', ...
        'xdata', [4 2 8], ...
        'ydata', [7 1 6]));
A(2).data = [9 3 2; 7 6 5];
A(2).nest.testnum = 'Test 2';
A(2).nest.xdata = [3 4 2];
A(2).nest.ydata = [5 0 9];

A
```

结构体的常用函数

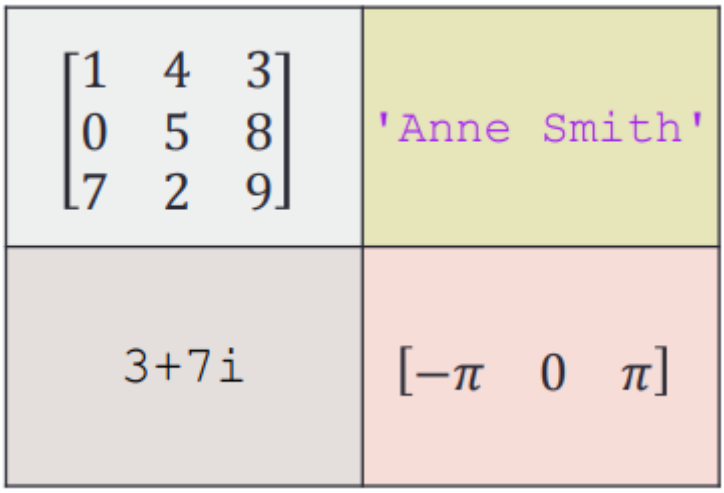
函数	作用
`struct`	创建结构体
`struct2cell`	将结构体转换为元胞数组
`cell2struct`	将元胞数组转换为结构体
`isstruct`	判断某变量是否是结构体
`structfun`	对结构体的每个字段都应用某函数
`fieldnames`	获取结构体的所有字段名
`isfield`	判断结构体是否包含某字段
`getfield`	获取结构体某字段的值
`setfield`	为结构体中的某字段赋值
`rmfield`	删除结构体中的某字段
`orderfields`	为结构体字段排序

元胞数组(cell)

在MATLAB中,元胞数组是一个可以容纳不同类型元素的数据结构,类似于Python语言中的列表.

元胞数组的基本使用

- 我们可以使用`{}`像定义矩阵一样定义元胞数组:




```
A = { [1 4 3; 0 5 8; 7 2 9]     'Anne Smith' ;...
3+7i         -pi:pi:pi}
```

```
A(1,1)={[1 4 3; 0 5 8; 7 2 9]};
A(1,2)={'Anne Smith'};
A(2,1)={3+7i};
A(2,2)={-pi:pi:pi};
A
```

```
A{1,1}=[1 4 3; 0 5 8; 7 2 9];
A{1,2}='Anne Smith';
A{2,1}=3+7i;
A{2,2}=-pi:pi:pi;
A
```

上面三种方式是等价的,其中第二种方式使用**单元索引**赋值,而第三种方式使用**内容索引**赋值.

- 有两种方式访问元胞数组中的数据,分别是: **单元索引** `()` 和**内容索引** `{}`.

  因为元胞数组的子集仍为元胞数组,在索引器内容的使用,我们有必要指明我们要访问的的是一个**子元胞数组**还是**元胞数组对应区域中的内容**.

  - 使用**单元索引** `()`,我们得到的是一个**子元胞数组**.
  - 使用**内容索引** `{}`,我们得到的是**元胞数组对应区域中的内容**.

  关于单元索引和内容索引的区别,请参考官方文档

元胞数组的常用函数

函数	作用
`cell`	创建一个元胞数组
`iscell`	判断某变量是否为元胞数组
`cell2mat`	将元胞数组转为矩阵
`cell2struct`	将元胞数组转为结构体
`mat2cell`	将数组转换为指定大小元胞数组
`num2cell`	将数组转换为相同大小的元胞数组
`struct2cell`	将结构体转换为元胞数组
`celldisp`	递归显示元胞数组中的内容
`cellplot`	以图像形式绘制元胞数组的结构
`cellfun`	对元胞数组的每个元胞应用某函数

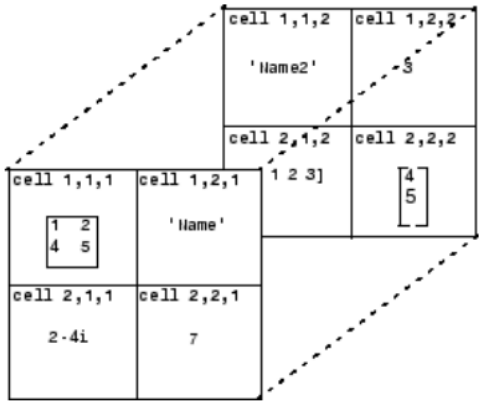
其中`mat2cell`函数可以在转换的时候指定元胞数组各元胞的尺寸.

```
a = magic(3)

b = num2cell(a)
% 得到
% [8] [1] [6]
% [3] [5] [7]
% [4] [9] [2]

c = mat2cell(a, [1 2], [2, 1])
% 得到
% [1x2 double] [6]
% [2x2 double] [2x1 double]
```

高维元胞数组

一个三维的元胞数组可以有**行**(row),**列**(column),**层**(layer)三个维度.在对元胞数组进行索引时,优先级从高到低的顺序分别是: 行→列→层.

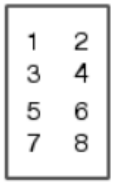
```
A{1,1,1} = [1 2;4 5];
A{1,2,1} = 'Name';
A{2,1,1} = 2-4i;
A{2,1,1} = 7;
A{1,1,2} = 'Name2';
A{1,2,2} = 3;
A{2,1,2} = 0:1:3;
A{2,2,2} = [4 5]';
```



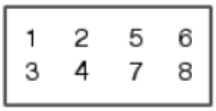
使用 `cat` 函数可以在指定维度上对元胞数组进行拼接.

```
A=[1 2;3 4]; B=[5 6;7 8];
```

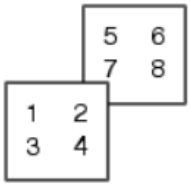
```
C=cat(1,A,B)
```

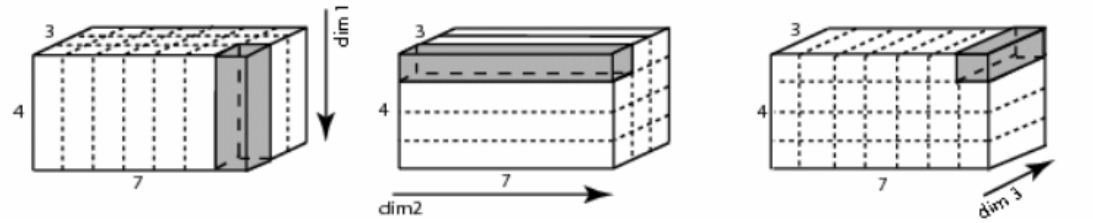



```
C=cat(2,A,B)
```




```
C=cat(3,A,B)
```





判断变量数据类型的函数

下列函数可以对变量类型进行判断:

函数	作用
`isinteger`	判断输入参数是否为整型数数组
`islogical`	判断输入参数是否为逻辑量数组
`isnumeric`	判断输入参数是否为数值数组
`isreal`	判断输入参数是否为实数数组
`ischar`	判断输入参数是否为字符数组
`iscell`	判断输入参数是否为元胞数组
`isfloat`	判断输入数组是否为浮点数组
`ishandle`	判断输入数组是否有效的图形句柄
`isempty`	判断输入数组是否为空
`isprime`	确定哪些数组元素为质数
`isnan`	确定哪些数组元素为 `NaN`
`isinf`	确定哪些数组元素为 `Inf`
`isequal`	判断数组是否相等

文件读写

MATLAB支持的文件类型如下:

文件内容	扩展名	读取文件的函数	写入文件的函数
MATLAB数据	`*.mat`	`load`	`save`
Excel表格	`*.xls`,`*.xlsx`	`xlsread`	`xlswrite`
空格分隔的数字	`*.txt`	`load`	`save`

读写MATLAB格式的数据

MATLAB工作区内的数据可以以`*.mat`格式保存在文件中.使用`save`函数将数据存入文件,使用`load`函数从文件中读取数据.

- `save`函数的语法如下:
  - `save(filename,variables)`将变量`variables`以二进制形式存入文件中.
  - `save(filename,variables,'-ascii')`将变量`variables`以文本形式存入文件中.
- `load`函数的语法如下:
  - `load(filename)`从二进制形式文件中读取数据.
  - `load(filename,'-ascii')`从文本形式文件中读取数据.

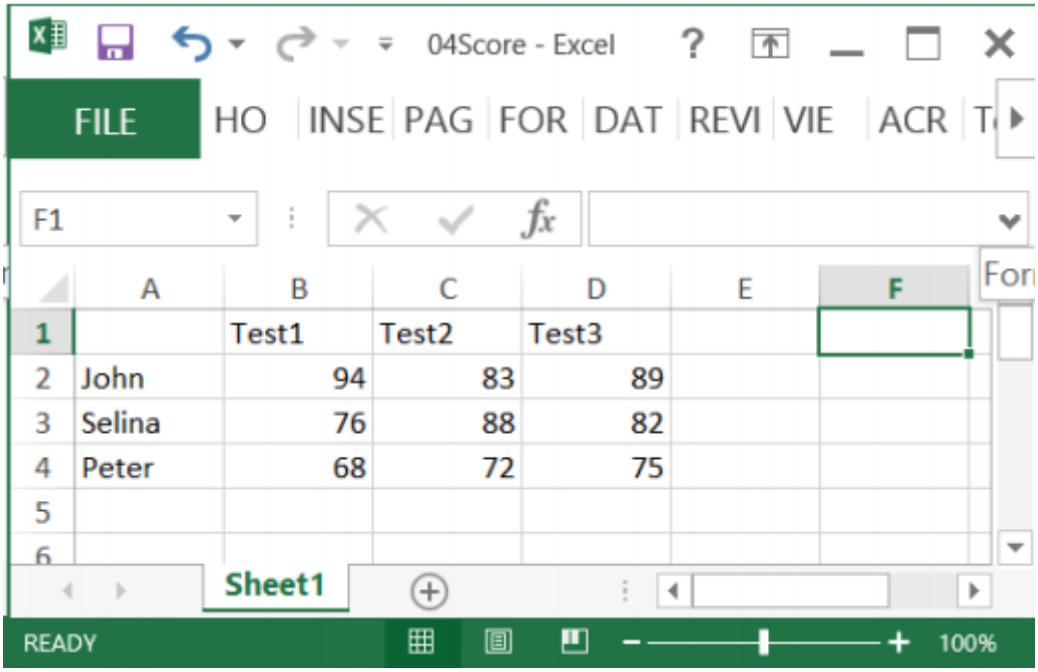
其中参数`filename`和`variables`都是字符串格式,若不指定`variables`参数,则将当前工作区内所有变量存入文件中.

> 复杂的数据格式,如`struct`和`cell`,不支持以二进制格式存储.

读写Excel表格

使用`xlsread`和`xlswrite`函数可以读写Excel数据,语法如下:

- 读取Excel文件的语法:`[num,txt,raw] = xlsread(filename,sheet,xlRange)`




```
Score = xlsread('04Score.xlsx')
Score = xlsread('04Score.xlsx', 'B2:D4')
[Score Header] = xlsread('04Score.xlsx')
```

- 写入Excel的语法:`xlswrite(filename,A,sheet,xlRange)`

```
M = mean(Score);
xlswrite('O4Score.xlsx', M, 1, 'E2:E4');
xlswrite('O4Score.xlsx', {'Mean'}, 1, 'E1');
```