

Storing Values for Safe Keeping

Variables and their Values

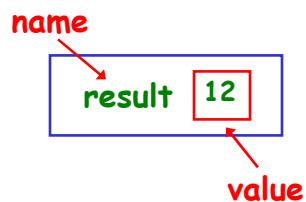


CS112 Scientific Computation

Department of Computer Science
Wellesley College

Variables

- We store information in **variables** that have two parts:



- Value can be a **number** or **string of characters**
- Values are assigned to variables using an **assignment statement**

```
>> result = 12
```

Assignment statements

- General format of an assignment statement:

`variable-name = expression`

- Right hand side can be any expression that evaluates to a value:

```
>> hypotenuse = sqrt(3^2 + 4^2)
```

- Executing an assignment statement:
 - 1) **evaluate the expression** on the right side
 - 2) **assign value** of expression to variable named on the left side

Variables 2-3

Variables inside expressions

- Inside an expression, we can refer to the value of a variable by its name

```
>> (hypotenuse^2 - 5) / 2
```

- Assignment statements using variables on the right side are no problem, provided they have a value

```
>> age = 20
```

```
>> happyBirthday = age + 1
```

age	<input type="checkbox"/>
happyBirthday	<input type="checkbox"/>

Variables 2-4

Not your typical algebra

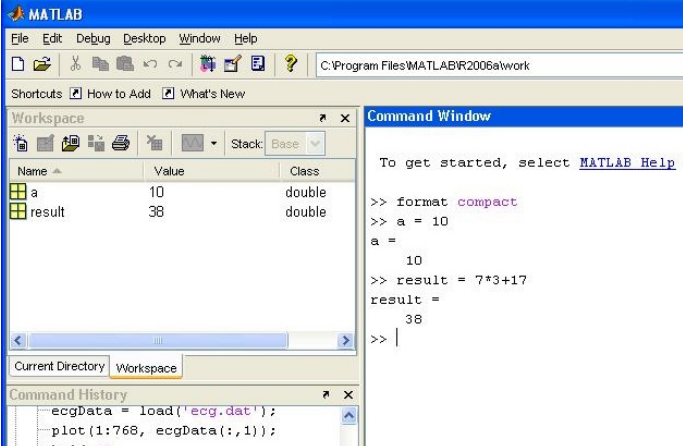
- The same variable name can appear on both sides of an assignment statement:

```
>> age = 20  
>> happyBirthday = age + 1  
>> age = age - 12
```

age	<input type="checkbox"/>
happyBirthday	<input type="checkbox"/>

Variables 2-5

Variables and the MATLAB Workspace



The screenshot shows the MATLAB environment. The **Workspace** window displays the following variables:

Name	Value	Class
a	10	double
result	38	double

The **Command Window** shows the following commands and output:

```
>> format compact  
>> a = 10  
a =  
    10  
>> result = 7*3+17  
result =  
    38  
>> |
```

The **Command History** window shows the following commands:

```
ecgData = load('ecg.dat');  
plot(1:768, ecgData(:,1));  
hold on
```

Variables 2-6

Hunting for 's

```
>> x = 17  
>> x + 1 = 10  
>> x = x / x + x  
>> diff = (x - y) / 2  
>> end = x - 18  
>> y = x = 2
```

Variables 2-7

Assigning variables in a script

- Statements in script are executed as if they were typed directly into the Command Window
- Variables created in CW can be changed by assignments in scripts and vice versa!

```
% assignVars.m  
% assigns variables  
% to values
```

```
a = 1  
b = 2  
c = 3
```



Variables 2-8

Speaking of 's

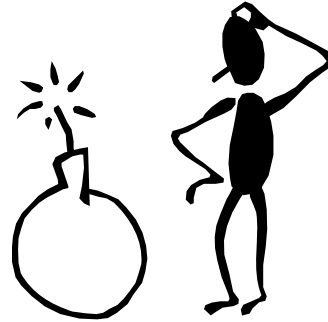
- What would happen if the script contained a **syntax error**? For example:

```
% buggyScript.m
```

```
a = 10
```

```
b = 20 +
```

```
c = 30
```



Variables 2-9

What's in a name?

- **Variable names** must start with a letter and may contain any number of letters, digits and underscore characters ('_')

```
C3PO    my_monthly_pay    yourTurn
```

- Case matters!

```
sohie ≠ Sohie
```

- MATLAB reserves some words for special purposes:

```
break case catch continue else elseif end  
for function global if otherwise persistent  
return switch try while
```

Variables 2-10

Choose concise, meaningful names

- The good, the bad, & the ugly

`maxArea`
`xyyyz_3b`
`totalMonthlyHedgeFunds`
`2pi`
`tf`
`result`

- Case matters!

`sohie` ≠ `Sohie`



Variables 2-11

Time-out exercise

- Write a sequence of assignment statements that exchange the values of `thing1` & `thing2`

- Before

<code>thing1</code>	2
<code>thing2</code>	5

- After

<code>thing1</code>	5
<code>thing2</code>	2



Variables 2-12

Floating point numbers

- Decimal numbers are represented as a type called **double**

```
>> number = 8.43  
number =  
    8.4300
```

- MATLAB normally prints 4 digits after the decimal point and rounds numbers up or down to fit*



* Compact notation is used only for printing - try typing **format long**

Variables 2-13

Really big & really little numbers

- MATLAB uses scientific notation to print very large or very small numbers

```
>> number = 1230000000  
number =  
    1.2300e+009  
>> number = 0.000789  
number =  
    7.8900e-004
```

- MATLAB has a special representation for really big numbers

```
>> number = 1.2e+9999  
number =  
    Inf
```


* What do get when you type: **1/0** or **1/inf** or **0*inf**?

Variables 2-14

Strings

- **Strings** consist of letters, digits, symbols, and white space, surrounded by single quotes:

```
myName = 'Sam I am'  
eec = ' )&it:s;elf,'  
thirteen = '6 + 7'
```

- Common  's

```
callMe = Ishmael  
reply = 'Madam, I'm Adam'
```



Variables 2-15

Unfriendly programs

- The following program converts pounds to stones:

```
% weights.m  
pounds = input(' ');  
stones = 0.0714 * pounds
```

- If we run it by typing

```
>> weights
```

it would just sit there and look at us



Variables 2-16

Friendly programs

- Tell the user what's needed and what's printed:

```
% weights.m  
pounds = input('Enter your weight in pounds: ');  
disp('Your weight in stones is: ');  
stones = 0.0714 * pounds
```

- When run...

```
>> weights  
Enter your weight in pounds: 120  
Your weight in stones is:  
stones =  
    8.5680
```

