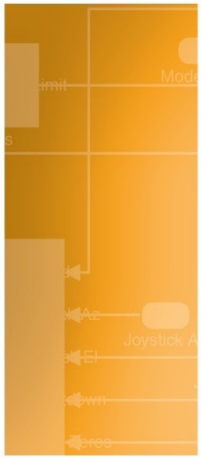
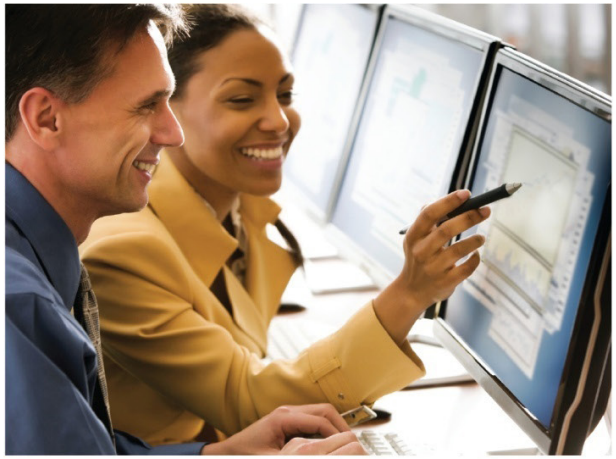
Increasing Automation with Programming Constructs

MATLAB® Fundamentals for Aerospace Applications

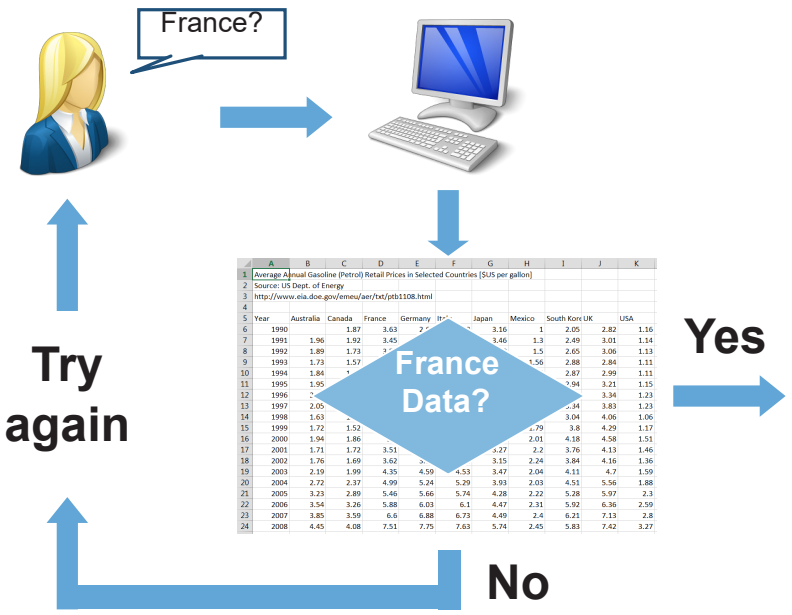




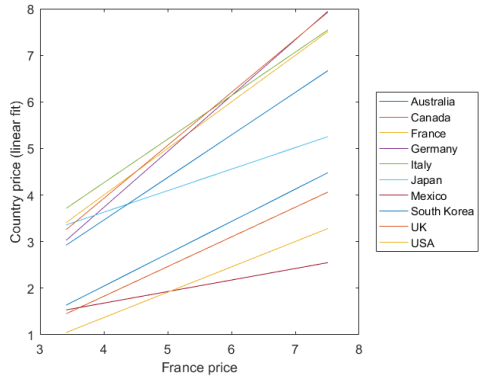
MathWorks | Training Services

Outline

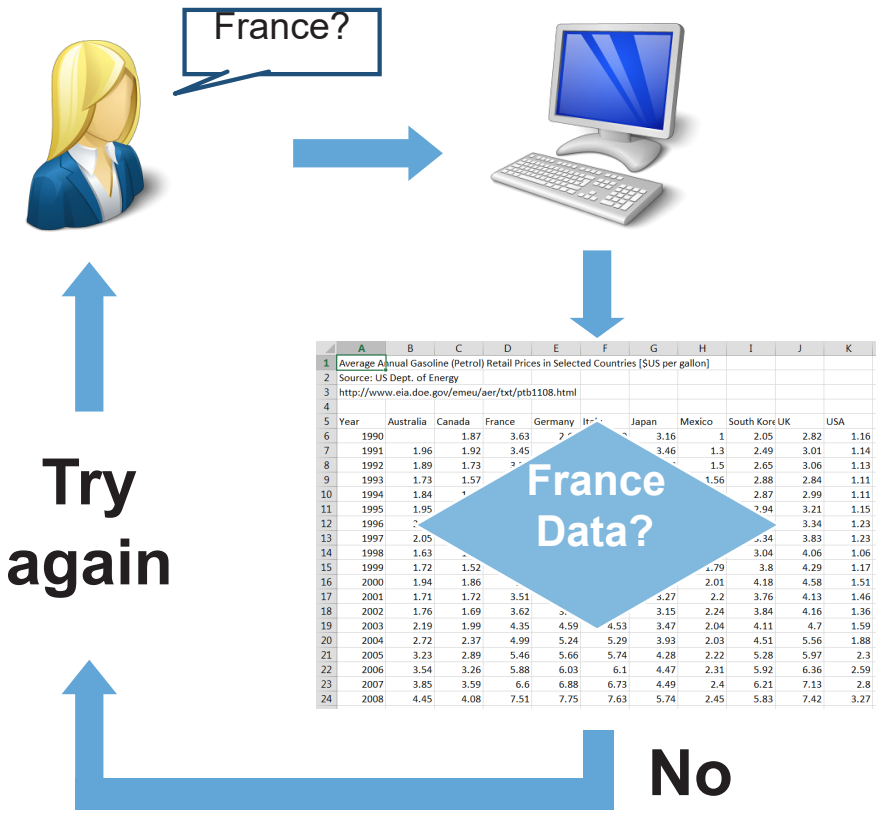
- Programming constructs
- User interaction
- Decision branching
- Loops




```
Australia/France slope: 0.69351
Canada/France slope: 0.63727
France/France slope: 1
Germany/France slope: 1.1991
Italy/France slope: 0.93439
Japan/France slope: 0.4621
Mexico/France slope: 0.24761
South Korea/France slope: 0.91372
UK/France slope: 1.1382
USA/France slope: 0.54434
```



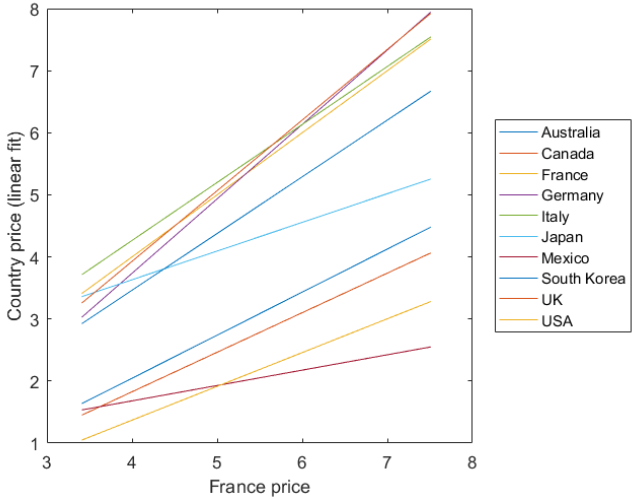
Course Example: Comparing Prices



Average Annual Gasoline (Petrol) Retail Prices in Selected Countries ($US per gallon)
Source: US Dept. of Energy
http://www.eia.doe.gov/emeu/aer/txt/ptb1108.html

Year	Australia	Canada	France	Germany	Japan	Mexico	South Kore	UK	USA
1990		1.87	3.63		3.16	1	2.05	2.82	1.16
1991	1.96	1.92	3.45		3.46	1.3	2.49	3.01	1.14
1992	1.89	1.73				1.5	2.65	3.06	1.13
1993	1.73	1.57				1.56	2.88	2.84	1.11
1994	1.84						2.87	2.99	1.11
1995	1.95						2.94	3.21	1.15
1996								3.34	1.23
1997	2.05						3.34	3.83	1.23
1998	1.63						3.04	4.06	1.06
1999	1.72	1.52				1.79	3.8	4.29	1.17
2000	1.94	1.86				2.01	4.18	4.58	1.51
2001	1.71	1.72	3.51		3.27	2.2	3.76	4.13	1.46
2002	1.76	1.69	3.62		3.15	2.24	3.84	4.16	1.36
2003	2.19	1.99	4.35	4.59	4.53	3.47	2.04	4.11	4.7	1.59
2004	2.72	2.37	4.99	5.24	5.29	3.93	2.03	4.51	5.56	1.88
2005	3.23	2.89	5.46	5.66	5.74	4.28	2.22	5.28	5.97	2.3
2006	3.54	3.26	5.88	6.03	6.1	4.47	2.31	5.92	6.36	2.59
2007	3.85	3.59	6.6	6.88	6.73	4.49	2.4	6.21	7.13	2.8
2008	4.45	4.08	7.51	7.75	7.63	5.74	2.45	5.83	7.42	3.27

```
Australia/France slope: 0.69351
Canada/France slope: 0.63727
France/France slope: 1
Germany/France slope: 1.1991
Italy/France slope: 0.93439
Japan/France slope: 0.4621
Mexico/France slope: 0.24761
South Korea/France slope: 0.91372
UK/France slope: 1.1382
USA/France slope: 0.54434
```



User Interaction

Text	`input`	`disp` `fprintf`	`warning` `error`

Graphical	`inputdlg`
	`listdlg`
	`uigetfile`
	
	`ginput`
		`msgbox`
		`waitbar`
			`warndlg`
			`errordlg`

Decision Branching

Yes

No

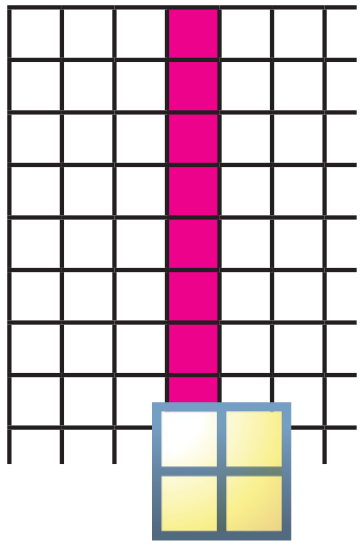
```
myctry = inputdlg('Enter a country:');
idx = strcmpi(myctry,country);

if any(idx)
    myPrices = Prices(:,idx);
    if ~strcmp(myctry,country{idx})
        warning('Modified country name')
        myctry = country{idx};
    end
    plot(Year,myPrices,'o-')
    xlabel('Year')
    ylabel([myctry,' prices'])
else
    error(['No data for ',myctry])
end
```

Mutually exclusive options

For-Loops

```
for k = 1:length(country)
    c = polyfit(myPrices,Prices(:,k),1);
    disp([country{k},'/',myctry,': ',num2str(c(1))])
end
```

```
Australia/France slope: 0.69351
Canada/France slope: 0.63727
France/France slope: 1
Germany/France slope: 1.1991
Italy/France slope: 0.93439
Japan/France slope: 0.4621
Mexico/France slope: 0.24761
South Korea/France slope: 0.91372
UK/France slope: 1.1382
USA/France slope: 0.54434
```



Determining Size

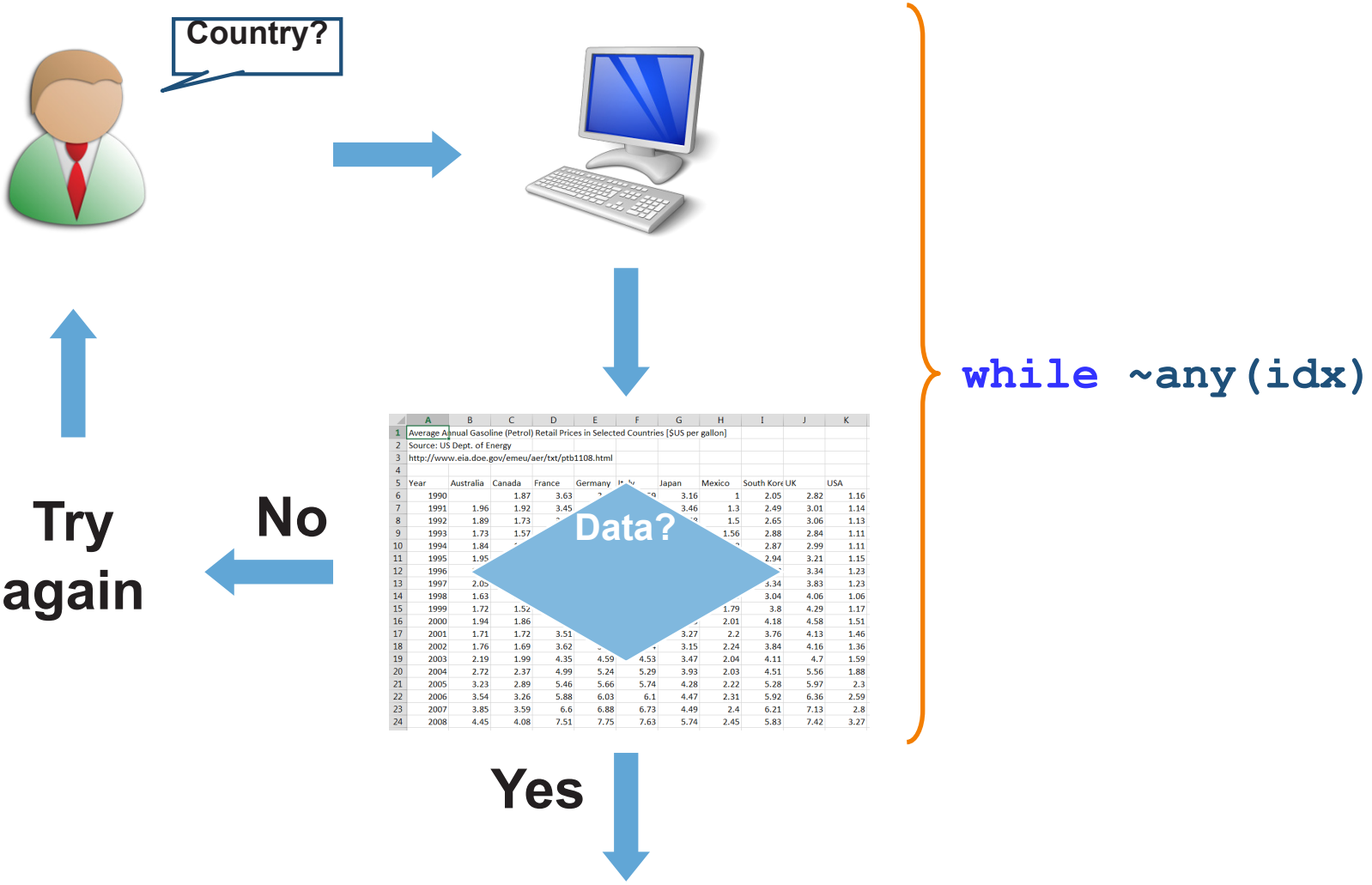
m-by-*n*

n-by-1

1-by-*n*

`size(x)`	[*m n*]	[*n* 1]	[1 *n*]
`size(x,1)`	*m*	*n*	1
`size(x,2)`	*n*	1	*n*
`length(x)`	*max(m,n)*	*n*	*n*
`numel(x)`	*m*n*	*n*	*n*

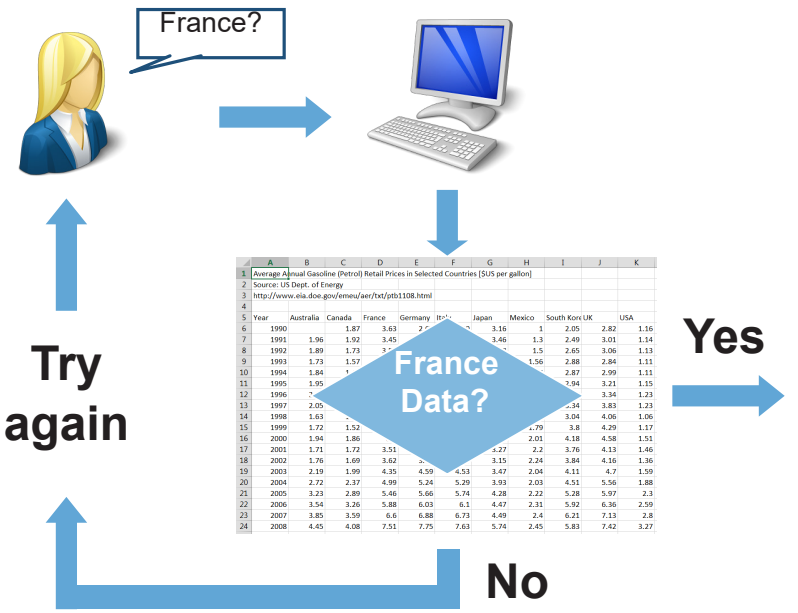
While-Loops



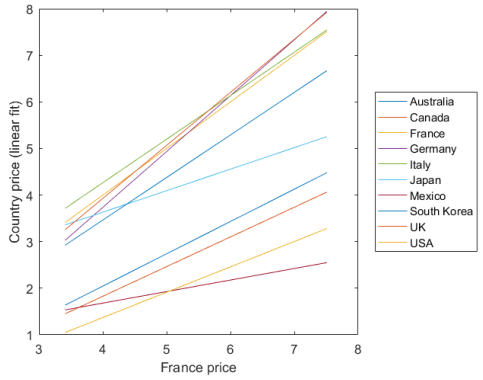
	A	B	C	D	E	F	G	H	I	J	K
1	Average Annual Gasoline (Petrol) Retail Prices in Selected Countries ($US per gallon)										
2	Source: US Dept. of Energy										
3	http://www.eia.doe.gov/emeu/aer/txt/ptb1108.html										
4											
5	Year	Australia	Canada	France	Germany	[illegible]	Japan	Mexico	South Kore	UK	USA
6	1990		1.87	3.63	[illegible]	[illegible]	3.16	1	2.05	2.82	1.16
7	1991	1.96	1.92	3.45	[illegible]	[illegible]	3.46	1.3	2.49	3.01	1.14
8	1992	1.89	1.73	[illegible]	[illegible]	[illegible]	[illegible]	1.5	2.65	3.06	1.13
9	1993	1.73	1.57	[illegible]	[illegible]	[illegible]	[illegible]	1.56	2.88	2.84	1.11
10	1994	1.84	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	2.87	2.99	1.11
11	1995	1.95	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	2.94	3.21	1.15
12	1996	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	3.34	1.23
13	1997	2.05	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	3.34	1.23
14	1998	1.63	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	3.04	4.06	1.06
15	1999	1.72	1.52	[illegible]	[illegible]	[illegible]	[illegible]	1.79	3.8	4.29	1.17
16	2000	1.94	1.86	[illegible]	[illegible]	[illegible]	[illegible]	2.01	4.18	4.58	1.51
17	2001	1.71	1.72	3.51	[illegible]	[illegible]	3.27	2.2	3.76	4.13	1.46
18	2002	1.76	1.69	3.62	[illegible]	[illegible]	3.15	2.24	3.84	4.16	1.36
19	2003	2.19	1.99	4.35	4.59	4.53	3.47	2.04	4.11	4.7	1.59
20	2004	2.72	2.37	4.99	5.24	5.29	3.93	2.03	4.51	5.56	1.88
21	2005	3.23	2.89	5.46	5.66	5.74	4.28	2.22	5.28	5.97	2.3
22	2006	3.54	3.26	5.88	6.03	6.1	4.47	2.31	5.92	6.36	2.59
23	2007	3.85	3.59	6.6	6.88	6.73	4.49	2.4	6.21	7.13	2.8
24	2008	4.45	4.08	7.51	7.75	7.63	5.74	2.45	5.83	7.42	3.27

Summary

- Programming constructs
- User interaction
- Decision branching
- Loops




```
Australia/France slope: 0.69351
Canada/France slope: 0.63727
France/France slope: 1
Germany/France slope: 1.1991
Italy/France slope: 0.93439
Japan/France slope: 0.4621
Mexico/France slope: 0.24761
South Korea/France slope: 0.91372
UK/France slope: 1.1382
USA/France slope: 0.54434
```



Test Your Knowledge

1. If `x = -4` what will be the result of running the following code?

```
if (x < 0) || (sqrt(x) > pi)
    y = 7;
else
    y = 2;
end
```

   A. `y = 7;`
   B. `y = 2;`
   C. `y` will be left undefined
   D. An error message due to taking the square root of a negative number
   E. An error message due to comparing an imaginary number to a real number

Test Your Knowledge

2. What construction should you use to loop over a block of code an indefinite number of times?

   A. `if`

   B. `for`

   C. `switch`

   D. `while`

   E. Logical indexing