**MATLAB/Octave - Moving Data Around**

**# printing matrix A value for later use**

>> A

A =

1 2

3 4

5 6

**#shape of matrix A (former element shoes number of rows and latter one shows number of columns)**

>> size(A)

ans =

3 2

**#display number of rows in a given matrix A**

>> size(A, 1)

ans =

3

**#display number of columns in a given matrix A**

>> size(A, 2)

ans =

2

**#printing vector V for later use**

>> V

V =

1 2 3

**#length of vector V**

>> length(V)

ans =

3

**#length of matrix A, here length function will return either number of rows or columns whichever is maximum**

>> length(A)

ans =

3

**#to print the current working directory**

>> pwd

ans =

'/MATLAB Drive'

**#print the content of the current directory**

>> ls

machine-learning-ex machine-learning-ex.zip Published

**#change directory**

>> cd machine-learning-ex/

**#go back to the previous directory**

>> cd ..

**#print all variables present in the current session**

>> who

Your variables are:

A I V a ans b c v w

**#print all variables (detailed view) present in the current session**

>> whos

Name Size Bytes Class Attributes

A 3x2 48 double

I 3x3 72 double

V 1x3 24 double

a 1x1 8 double

ans 1x13 26 char

b 1x2 4 char

c 2x3 48 double

v 3x1 24 double

w 1x1000 8000 double

**#delete variable 'w' from the current session**

>> clear w

**#confirm that 'w' variable is not in the current session list**

>> whos

Name Size Bytes Class Attributes

A 3x2 48 double

I 3x3 72 double

V 1x3 24 double

a 1x1 8 double

ans 1x13 26 char

b 1x2 4 char

c 2x3 48 double

v 3x1 24 double

**#load data file present in the current directory**

>> load priceY.dat

**#print the content of the data file read in the above step, here variable name is same as filename without extension**

>> priceY

priceY =

2150

1250

4100

2563

6452

<.. more values>

**#create new variable from existing variable, assign top 10 elements of priceY vector to new variable**

>> w = priceY(1:10)

w =

2150

1250

4100

2563

6452

8452

6547

5124

7519

3520

**#save the vector/matrix in the new file**

>> save hello.mat w

**#to delete all the variables from the current session**

>> clear

**#check if everything is gone or not**

>> whos

**#load the saved file again**

>> load hello.mat

**#confirmed if the load is successful or not**

>> whos

Name Size Bytes Class Attributes

w 10x1 80 double

**#save the vector/matrix in the human readable format**

>> save hello.txt w -ascii

**#create matrix A for later use**

>> A = [1 2; 3 4; 5 6]

A =

1 2

3 4

5 6

**#fetch the specific element of A, in this case element from 3rd row 2nd column**

>> A(3,2)

ans =

6

**#fetch all element of specific row, in this case get all element from 2nd row**

>> A(2,:)

ans =

3 4

**#fetch all element of specific column, in this case get all element from 2nd column**

>> A(:,2)

ans =

2

4

6

**#fetch multiple rows element**

>> A([1,3],:)

ans =

1 2

5 6

**#assign new values on the specific location of the matrix, in this case assign [10;11;12] vector to the second row of the matrix**

>> A(:,2) = [10; 11; 12]

A =

1 10

3 11

5 12

**#concatenate 2 different matrix/vector one after another using comma or space**

>> A = [A, [100; 101; 102]]

A =

1 10 100

3 11 101

5 12 102

**#check the size of matrix A after above concatenation**

>> size(A)

ans =

3 3

**#put all elements in a single column vector using colon ':'**

>> A(:)

ans =

1

3

5

10

11

12

100

101

102

**#create matrix A for later use**

>> A = [1 2; 3 4; 5 6]

A =

1 2

3 4

5 6

**#create matrix B for later use**

>> B = [11 12; 13 14; 15 16]

B =

11 12

13 14

15 16

**#concatenate two matrix/vectors one after another using comma or space**

>> C = [A B]

C =

1 2 11 12

3 4 13 14

5 6 15 16

**#concatenate two matrix/vectors one below another using semicolon**

>> C = [A; B]

C =

1 2

3 4

5 6

11 12

13 14

15 16

**#check the size of matrix C after above concatenation**

>> size(C)

ans =

6 2