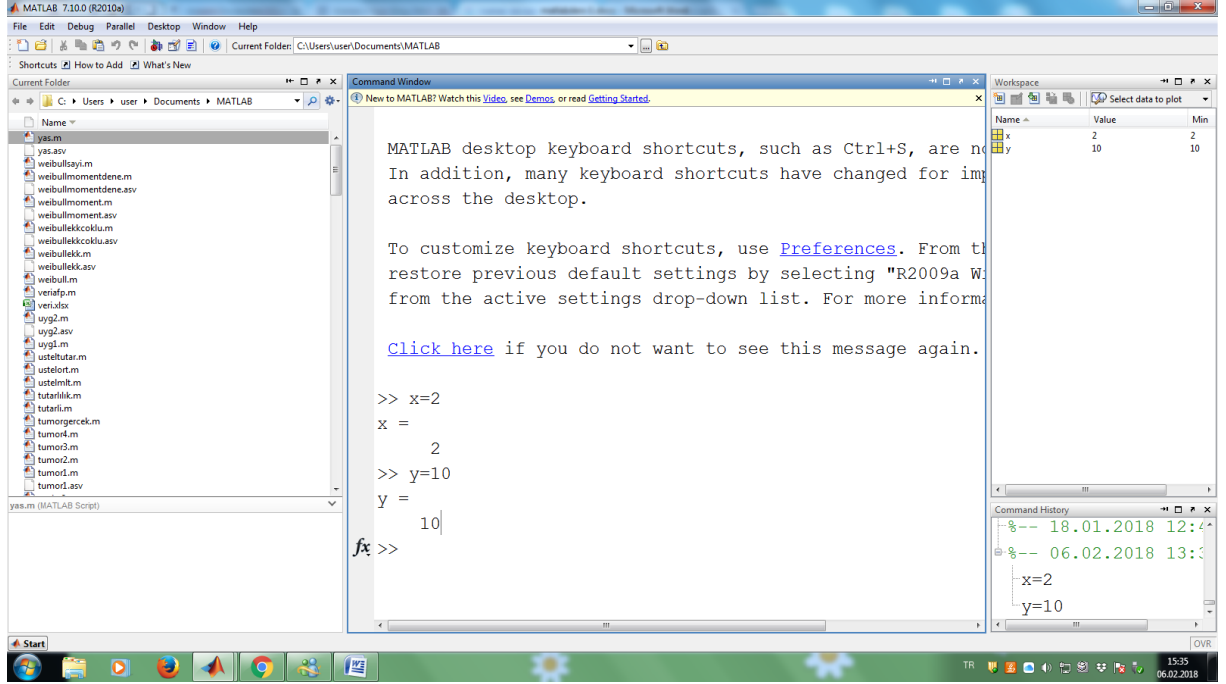


Matlab'ta Temel Matematiksel İşlemler

Matlab'da komut ekranına $x=2$ ve $y=10$ yazıp enter tuşuna bastığımızda göreceğimiz ekran görüntüsü aşağıdaki gibi olacaktır.



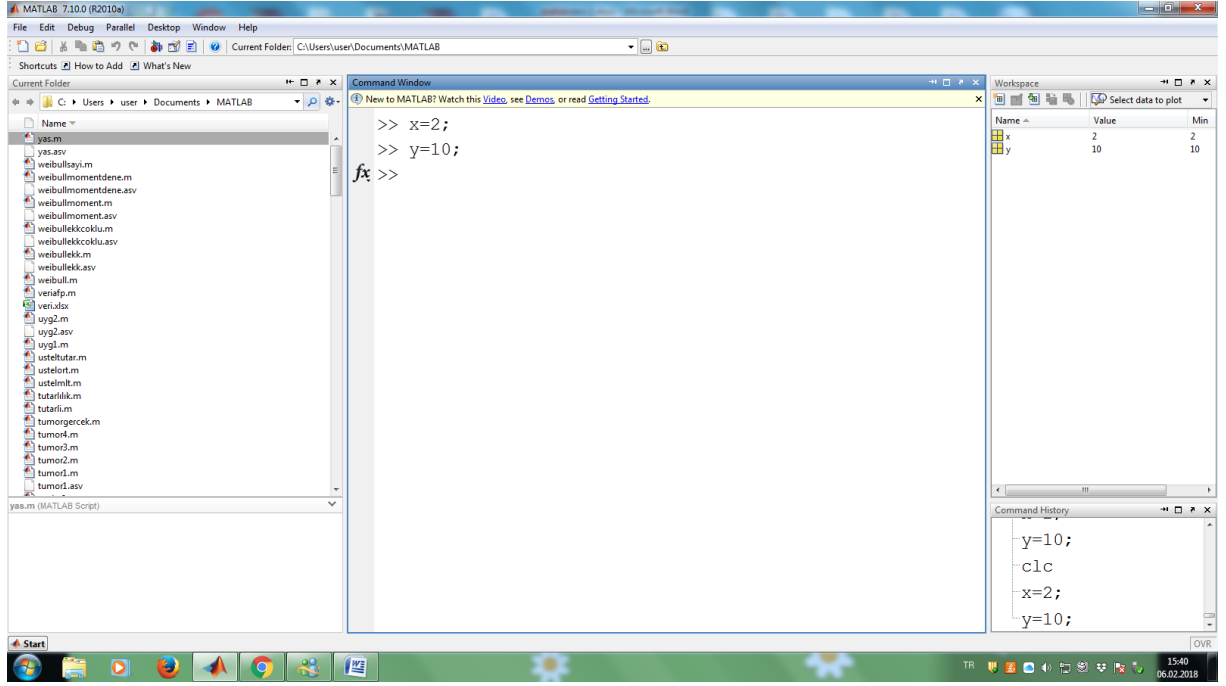
Matlab'da `clc` komutu ile ekran temizlenir. Tanımladığımız değişken değerlerinden sonra “;” yazılırsa değişkenlerin değerleri tekrar komut ekranında yazmayacaktır. Yani, `Clc` yazıp enter tuşuna basıldıktan sonra,

```
>>x=2;
```

ve

```
>>y=10;
```

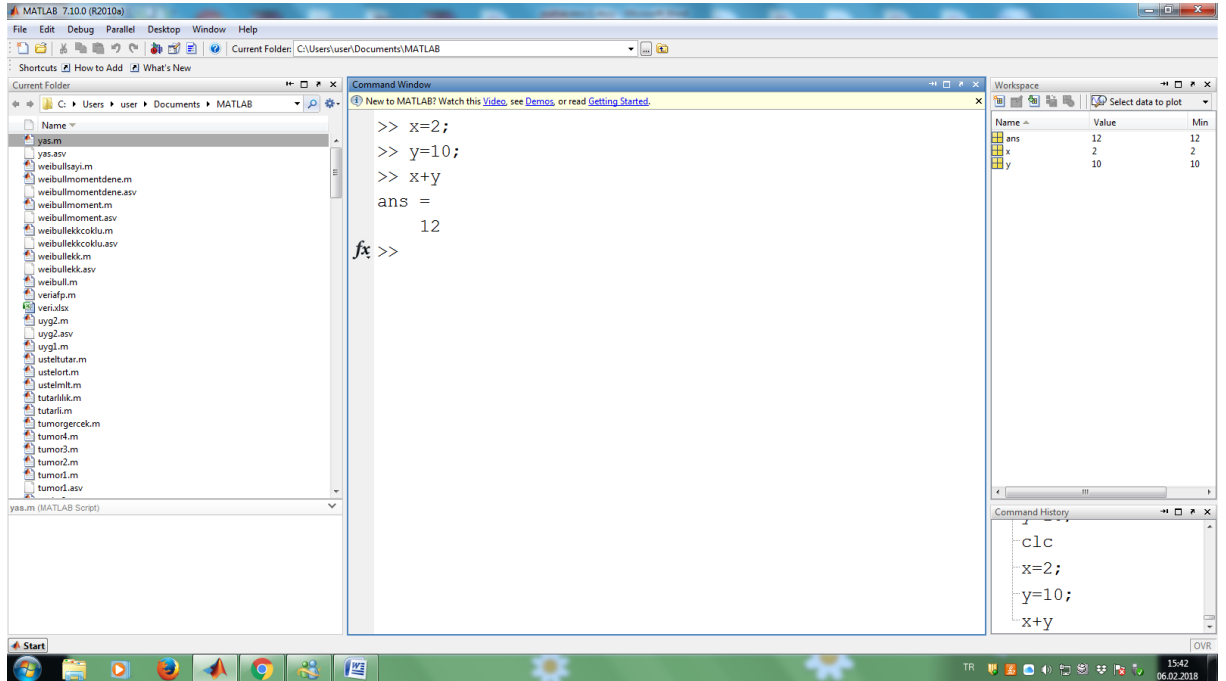
yazılırsa ekran aşağıdaki gibi olacaktır.



x değişkenine 2 değeri ve y değişkenine 10 değeri atanmış oldu. Komut ekranında,

>>x+y

yazılıp enter tuşuna basıldığında,



ekranı karşımıza çıkar. Burada, ans (**ans**wer kısaltılmışı) değişkeni ile işlemin sonucu tanımlanmaktadır. Buna göre, bir takım temel matematiksel işlem aşağıda verildiği gibi yapılır.

Eğer komut ekranına,

```
>>x+y;
```

yazılıysa sonuç yine ans değişkenine aktarılacaktı fakat ekrana sonucu yazmayacaktı. Bu durumda sonucu tekrar görmek için komut ekranına,

```
>> ans
```

yazıldığında

```
ans=
```

```
12
```

sonucunu verecekti. İstenilirse bu sonuçta yine bir değişkene aktarılabilir. O zaman,

```
>>z=x+y
```

denildiğinde

```
z=12
```

sonucunu ekrana yazar.

```
>>z=x+y;
```

yazıldığında sonucu z değişkenine aktarır ama ekrana yazmaz. z' nin içinde ne olduğu görülmek isteniyorsa ekrana

```
>>z
```

yazılır ve z değişkeninin değeri neyse o ekranda görüntülenir.

Matlab'da aritmetik işlemler yaptırmak için değişken tanımlamaya da gerek yoktur. Yapılmak istenilen işlem ekrana yazılıp enter tuşuna basıldığında Matlab aynı bir hesap makinesi gibi işlem görür. Örneğin,

```
>> 3+5
```

```
ans =
```

```
8
```

```
>> 7*8
```

```
ans =
```

```
56
```

```
>> 47/9
```

```
ans =
```

```
5.222222222222222
```

gibi.

```
>>x=2;
```

```
>> y=10;
```

```
>> z=x*y;
```

```
>> z
```

```
z =
```

```
20
```

```
>> size(z)
```

```
ans =
```

```
1 1
```

size() komutu ile parantez içerisinde yazılı değişkenin boyutu sorulur. Matlab tanımlanan tüm değişkenleri bir matris olarak düşünür. Bu nedenle, tek boyutlu olan bu değişkenin boyutunu 1x1 olarak verir.

```
>> x=[1 2 3 4 5 6];
```

```
>> x
```

```
x =
```

```
1 2 3 4 5 6
```

```
>> size(x)
```

```
ans =
```

```
1 6
```

x, 1sattır 6 sütünndan oluřan bir matristir. x' ile x matrisinin transpozu yazdırılır. Buna göre,

```
>> x'
```

```
ans =
```

```
1  
2  
3  
4  
5  
6
```

sonucu ekranda görülür.

```
>> t=x';
```

yazılrsa idi,

```
>> t
```

sorulduğunda ekranda

```
t =
```

```
1  
2  
3  
4  
5  
6
```

sonucu gözükürdü.

```
>> t1=t(1:3) yazıldığında
```

```
t1 =
```

```
1  
2  
3
```

sadece ilk üç elemanı ekrana yazar.

2x4 lük bir m matrisi tanımlansın;

```
>> m=[1 2 3 4;2 4 6 8]
```

```
m =
```

```
1 2 3 4
2 4 6 8
```

Bu matrisin transpozu m1 matrisine aktarılırsa,

```
>> m1=m';
```

```
>> m1
```

```
m1 =
```

```
1 2
2 4
3 6
4 8
```

olarak ekranda görünür.

```
>> m1(2,1)
```

```
ans =
```

```
2
```

```
>> m1(2,:)
```


```
ans =
```

```
2 4
```

```
>> m1(2,1)=9
```

```
m1 =
```

```
1 2
9 4
3 6
4 8
```



olur.

>> m1(1,:)=[] yazıldığında m1 matrisinin 1.satırda bulunan tüm elemanları silinir. Buna göre sonuç,

m1 =

```
9  4
3  6
4  8
```

olarak ekranda görünür.

>> m1(:,2) yazıldığında,

ans =

```
4
6
8
```

olur. Şimdi tekrar,

>> m1=m';

olarak alınsın ve işlemlere aşağıdaki gibi devam edilsin.

>>m2=m1*m (matris çarpımı)

m2 =

```
5  10  15  20
10  20  30  40
15  30  45  60
20  40  60  80
```

>>m3=m*m1

m3 =

```
30  60
60  120
```

Şimdi başka bir matris tanımlansın,

```
>> A=[5 7;6 9]
```

```
A =
```

```
5 7
```

```
6 9
```

```
>> det(A) (A matrisinin determinantını bulur)
```

```
ans =
```

```
2.9999999999999995
```

```
>> inv(A) (A matrisinin tersini bulur)
```

```
ans=
```

```
3.0000000000000005 -2.3333333333333338
```

```
-2.0000000000000004 1.6666666666666669
```

```
>> A^2 (Matrisin kendisi ile çarpımı-karesi)
```

```
ans =
```

```
67 98
```

```
84 123
```

```
>> A.^2 (Matrisin her bir elemanının karesi)
```

```
ans =
```

```
25 49
```

```
36 81
```

Matlab' ta tanımlı olan bazı özel matrisler vardır.

```
>> zeros(3,4)
```

```
ans =
```

```
0 0 0 0
```

```
0 0 0 0
```

```
0 0 0 0
```

zeros(n,m) - nxm lik (n satır, m sütunlu) sıfırlardan oluşan matris yaratır.

```
>> ones(2,3)
```


ans =

```
1 1 1
1 1 1
```

ones(n,m) - nxm lik (n satır, m sütunlu) birlerden oluşan matris yaratır.

```
>> eye(2)
```

ans =

```
1 0
0 1
```

eye(n) – n boyutlu birim matris yaratır.

```
>> magic(3)
```

ans =

```
8 1 6
3 5 7
4 9 2
```

magic(n)- nxn lik satır ve sütun toplamları birbirine eşit olan sihirli bir matris yaratır. Bunun gerçekte böyle olup olmadığı aşağıdaki gibi kontrol edilebilir.

```
>> m=magic(3)
```

m =

```
8 1 6
3 5 7
4 9 2
```

```
>> sum(m) (m matrisinin sütun toplamlarını verir)
```

ans =

```
15 15 15
```

```
>> sum(m') (m matrisinin transpozunun sütun toplamlarını verir- m matrisinin satır toplamları)
```

ans =

```
15 15 15
```

Matrislerde istenilen satır ve sütun elemanlarının toplatılmasını anlamak için aşağıdaki işlemlere bakılabilir.

```
>> A=[1 2 3;4 5 6;7 8 9]
```

```
A =
```

```
1 2 3
4 5 6
7 8 9
```

```
>> sum(A) (A nın tüm sütun elemanlarının toplamlarını verir)
```

```
ans =
```

```
12 15 18
```

```
>> sum(A(:,3)) (A nın 3. sütununda yer alan tüm satır elemanlarının toplamlarını verir)
```

```
ans =
```

```
18
```

```
>> sum(A(1:2,2:3)) (A nın 1. Satırından 2. Satırına kadar olan elemanları içinden  
2. Ve 3. sütuna ait olanlarının toplamlarını verir)
```

```
ans =
```

```
7 9
```

```
>> rank(A) (A matrisinin rankını hesaplar)
```

```
ans =
```

```
2
```