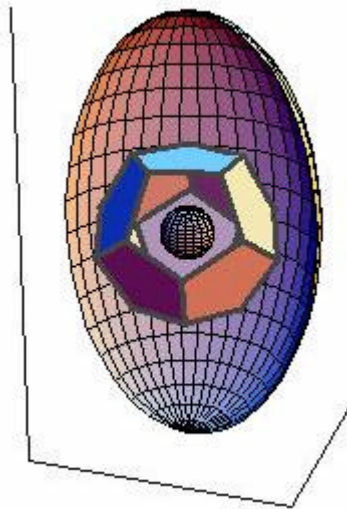


# Matlab-Übungen für den Einstieg

*„Learning by doing“*



## Inhaltsverzeichnis

---

Matlab-Übungen 1: Zahlenrechnen, arbeiten mit Variablen.....	3
Matlab-Übungen 2: Eingabe und arbeiten mit Vektoren .....	4
Matlab-Übungen 3: Eingabe und arbeiten mit Funktionen.....	5
Matlab-Übungen 4: 2D-Graphen.....	6
Matlab-Übungen 5: 3D-Graphen.....	7
Matlab-Übungen 6: Matrizen und Gleichungssysteme .....	8

---

**Probiere aus – das einfache Zeugs!**

© Rolf Wirz

Hinweis: Bei der Konzeption dieser Übungen ist das „Strickmuster“ des Matlab-Einführungsskripts ist das Matlab-Skript von M. J. Grote und Ch. Kirsch, Mathematisches Institut Universität Basel, 2003, als Vorbild beachtet worden. Vgl. <http://www.math.unibas.ch/~huber/IterVerf/MatlabIntroGroteKirsch.pdf> .

# Matlab-Übungen 1: Zahlenrechnen, arbeiten mit Variablen

Berechne jeweils mit Matlab die folgenden Ausdrücke oder gib diese Ausdrücke nach dem gezeigten Zeichen „>>“ in Matlab ein und **beschreibe dann in Worten**, was Matlab mit dem jeweiligen Ausdruck macht:

1. >> 4+7
2. >> v=4+7
3. >> v
4. >> v+2^3
5. >> clear all
6. >> v
7. >> v+2^3
8. >> c=sin(5\*pi/6)+1/3
9. >> c
- 10.>> format long
- 11.>> c
- 12.>> format short
- 13.>> c
- 14.>> d
- 15.>> d=c^2-5\*c+sqrt(2)-1
- 16.>> sqrt(-3)
- 17.>> b=3-4i
- 18.>> real(b)
- 19.>> imag(b)
- 20.>> abs(b)
- 21.>> round(24.3219)
- 22.>> round(24.9219)
- 23.>> round(24.9219)- 24.9219
- 24.>> round(round(24.9219)- 24.9219)
- 25.>> round(100\*24.9219)/100
- 26.>> help clear
- 27.>> f=5
- 28.>> f
- 29.>> clear f
- 30.>> f

## Matlab-Übungen 2: Eingabe und arbeiten mit Vektoren

Probiere aus und **beschreibe dann in Worten**, was Matlab mit dem jeweiligen Ausdruck macht:

1. `>> v1=[4 6 9 2]`
2. `>> v1(1)`
3. `>> v1(2)`
4. `>> v1(3)`
5. `>> v2=[2;1;3;5]`
6. `>> v2(1)`
7. `>> v2(4)`
8. `>> v2(1:2)`
9. `>> v2(2:4)`
10. `>> v2([1:4])`
11. `>> v2([1:3])`
12. `>> length(v2)`
13. `>> v1`
14. `>> v3=2*v1`
15. `>> v2`
16. `>> v4=2*v2`
17. `>> w1=v1+2`
18. `>> w2=v2+2`
19. `>> v1'`
20. `>> v2'`
21. `>> v1*v2`
22. `>> dot(v1,v2)`
23. `>> v1'*v2'`
24. `>> v2'*v1'`
25. `>> v3=[1 2 3]; v4=[4 5 6];`
26. `>> cross(v3,v4)`
27. `>> cross(v3',v4')`
28. `>> v1'*v2`
29. `>> v1*v2'`
30. `>> v1.*v2`
31. `>> sin(v1)`
32. `>> sin(v1')`
33. `>> u1=1:6`
34. `>> u2=3:6`
35. `>> u3=0:0.1:1`
36. `>> u4=0:0.3:2`
37. `>> u1*u1'`
38. `>> u1'*u1`
39. `>> clear all`

## Matlab-Übungen 3: Eingabe und arbeiten mit Funktionen

Probiere aus und **beschreibe dann in Worten**, was Matlab mit dem jeweiligen Ausdruck macht.

Klicke im Matlab oben links auf File → New → M-file, dann öffnet sich ein Editor.

- `function z=f2(x1,x2,x3)`  
`z=x1+x2^2+x3^2`

Speichere im Editor das Eingegebene unter dem Namen f2.m ab. Wichtig ist dabei, dass der File-Name mit dem Funktionsname übereinstimmt. Jetzt ist die Funktion gebrauchsfertig und kann im Matlab aufgerufen werden. Gib z.B. ein: danach

1. `>> f2(1,2,3)`

Schreibe analog wie oben die folgenden Zeichenketten in ein m-File mittels des Editors von Matlab (oben links auf File → New → M-file klicken):

- `function z = f(y); z = y^2 ;`

Speichere dann die Datei unter dem Namen f.m . Die Datei muss wie oben erwähnt denselben Namen haben wie die Funktion. Dann kann man die Funktion unter dem Namen f aufrufen:

Verfahre gleich mit der folgenden Zeichenkette:

- `function z1 = g(y1,y2)`  
`z1 = y1^2-exp(y2);`

f und g kann man dann wie vordefinierte Matlab-Funktionen benutzen :

1. `>> f(5)`
2. `>> [f(0) f(1) f(2) f(3) f(4) f(5)]`
3. `>> f([0 1 2 3 4]'``[0 1 2 3 4])`
4. `>> f([0 1 2 3 4])`
5. `>> f([0;1;2])`
6. `>> a=2; b=2; g(a,b)`
7. `>> g(b,a)`
8. `>> b=5; g(a^2,b^2)`
9. `>> f(sin(pi/2))`
10. `>> f2(f2(1,2,3),f2(2,3,4),f2(3,4,5))`
11. `>> sin(4)`
12. `>> tan(pi/2-0.1)`
13. `>> atan(pi/4)`
14. `>> exp(1)`

## Matlab-Übungen 4: 2D-Graphen

Probiere aus und **beschreibe dann in Worten**, was Matlab mit dem jeweiligen Ausdruck macht.

1. `>> x=0:0.01:2*pi;`
2. `>> plot(x,sin(x))`
3. `>> plot(x,sin(x),'-',x,cos(x),'-')`
4. `>> axis([-0.2 2*pi+0.2 -1.2 1.2])`
5. `>> legend('Sin','Cos')`
6. `>> xlabel('x')`
7. `>> ylabel('f(x)')`
8. `>> x=[1 2.5 3 4 1];`
9. `>> y=[1 -1 -2 1.5 0];`
10. `>> plot(x,y,'P')`
11. `>> axis([-2 5 -3 3])`
12. `>> % Erzeuge eigene Funktionen und tue es auch!`

## Matlab-Übungen 5: 3D-Graphen

Probiere aus und **beschreibe dann in Worten**, was Matlab mit dem jeweiligen Ausdruck macht.

1. >> x=-2:0.1:2;
2. >> y=-1:0.1:1;
3. >> [X,Y]=meshgrid(x,y);
4. >> Z=sin(X.\*X+Y.\*Y);
5. >> mesh(X,Y,Z)
6. >> xlabel('x')
7. >> ylabel('y')
8. >> zlabel('z')
9. >> Z=sin(X.\*X-Y.\*Y);
- 10.>> mesh(X,Y,Z)
- 11.>> xlabel('x')
- 12.>> ylabel('y')
- 13.>> zlabel('z')
- 14.>> % Erzeuge eigene Funktionen und tue es auch!

## Matlab-Übungen 6: Matrizen und Gleichungssysteme

Probiere aus und **beschreibe dann in Worten**, was Matlab mit dem jeweiligen Ausdruck macht.

1. `>> A=[1 2 4;2 5 3;6 9 9]`
2. `>> det(A)`
3. `>> A=[1 2 3;4 5 6;7 8 9]`
4. `>> det(A)`
5. `>> size(A,1)`
6. `>> size(A,2)`
7. `>> B=[0 1 0;0 0 1;1 0 0]`
8. `>> A+B`
9. `>> A*B`
10. `>> A.*B`
11. `>> A^2`
12. `>> B^3`
13. `>> A'`
14. `>> B'`
15. `>> A(1,2)`
16. `>> B(2,1)`
17. `>> A(2,1:3)`
18. `>> A(1:3,3)`
19. `>> A(1:2,2:3)`
20. `>> C=A([1 3],[1 3])`
21. `>> C*A(1:2,2:3)`
22. `>> A(1:2,2:3)* C`
23. `>> b=rand(3,1)`
24. `>> x=B\b`
25. `>> B*x`
26. `>> E=ones(3,2)`
27. `>> F=eye(3)`
28. `>> D=zeros(2,3)`
29. `>> eye(2)/C`
30. `>> G=eye(2)/C`
31. `>> G*C`
32. `>> C*G`
33. `>> G=[C D;E F]`