Matlab-quick reference sheet (*Pekka Alestalo/Heikki Apiola*)

1. `help, doc`
2. Use semicolon (;) to prevent display, WATCH OUT for extensive outputs. (CTR-C in the command window may help.) Write the name of your variable (without semicolon) to see its contents.
   If the variable is large, choose a smaller part like: `A(1:10,1:10)`
   Or use the **variable editor**. For that, click the "variable" icon in the "workspace-window".
3. The result of the previous unassigned command is in the variable `ans`. It will be ovewritten by the next command. **Recommendation:** Usually assign results to named variables: `name = ...;`
4. The uparrow key (↑) let's you scroll and edit the previous commands.
5. `format long` : Display max nr. of decimal digits (16). Doesn't affect accuracy of computation.
   `format rational`
   `format short`: Back to default
   `format compact`: Less empty lines in output display.
6. Create matrix: `A=[2 4 3;0 1 -1;3 5 7]`.
   Create row vector: `v=[1 2 3]` or column vector: `v=[1;2;3]`
7. Matrix product: `C=A*B`, A is $m \times p$, B is $p \times n$, C is $m \times n$
   Array product `C=A.*B`, generally: `A .op B`, same sizes or one is scalar.
8. Matrix operations dominate in general (Matrix laboratory). `A*B, A^p` .
   Array operations (pointwise) are `(.*), (.^),(./)`
   `u=[1 2 3], v=[-2 -2 -2], u.*v  u.^3` (long form: `u.^(3*ones(size(u)))`)
9. Transpose: `A'` (Conjugates also if complex entries).
10. Creating vectors (other than listing): `1:10`, `1:2:20`, `1:-0.1:-1`
    `linspace(a,b,N)`, default `N=100`.
    Column vector by transpose.
11. Indexing:
    `A(i,j)` Entry (i,j).
    `A(2,:)` Second row.
    `A(:,3)` Third column.
    `A(1:4,1:4)` rows 1:4, columns 1:4
    Update part of matrix:
    `A(1:4,1:4)=ones(4,4)` or for instance:
    `A(2,:)=A(2,:)-2*A(:,1)` (Gauss's row operation).
12. Concatenation of matrices: If A and B have equal nr. of rows, you can catenate side by side
    `[A B]` (or `[A, B]`).
    If equal nr.of columns, then on top of each other: `[A;B]`
13. plot:
    Create x-vector as discretized x-axis, compute $y = f(x)$ for function f,
    `plot(x,y);`
    Example: `x=linspace(-1,1,60); y=x.^2; plot(x,y)`
14. 3d-plot: on a grid of xy-plane, obtained by `meshgrid`-command.
    Example: $f(x, y) = \sin x \cos y$ on the square: $[-\pi, \pi] \times [-2\pi, 2\pi]$.

```
>> x=linspace(-pi,pi,25);
>> y=linspace(-2*pi,2*pi,50);
>> [X,Y]=meshgrid(x,y);
>> Z=cos(X).*sin(Y);
>> mesh(x,y,Z)    % Wireframe
>> surf(x,y,Z)    % Surface (also surfl, surfc, colorbar,...)
>> contour(x,y,Z) % Contours (also contourf, ...)
```

15. **Function handle** Let $f = (x, y) \to \cos(x)\sin(y)$
    Define "function handle":
    `f=@(x,y) cos(x).*sin(y)` , x and y have to be same size. In the previous one can do: `Z=f(X,Y)`
    f gets "at (x,y) the value cos(x)sin(y)"