Basics:
- SHIFT-ENTER = “compute”
- ( ) = grouping, [ ] = function argument, { } = table/list item markers
- Pre-defined functions usually begin with a capital: e.g. Sin[2]
- N[ … ] or … //N for numeric approximation
- % = “previous”, % number = “refer to line number”
- Use the palette (Tab switches fields)
- Control-C and Control-V copy and paste
- formatting output: Expand[ ], Simplify[ ], and FullSimplify[ ]
- Enter plain text via Format…Style…Text under horizontal line
- Help Index
- Open new notebook, cut, and paste for neatness

Useful Functions:
- Plot[ expression, {variable, domain min, domain max}] 

  Add ins: PlotRange _ { range min, range max}
  AxesLabel _ {“x title”, “y title”}
  PlotLabel _ “Title”

- Solve[ something = = something else, variable]
  and similarly for NSolve

- Series[ expression to expand, {variable, expand about, up to order}] 

Matrices:
- format is {{11, 12, 13}, {21, 22, 23}, {31, 32, 33}}
- matrix multiplication is “period” not “*”
- Det[ ], Inverse[ ], Eigenvalues[ ], Eigenvectors[ ]
Differential Equations:

- apostrophe is “prime”
- DSolve[y'[x] == 3*y[x], y[x], x] will solve (dy/dx)=3y for y as a function of x

- with initial condition: DSolve[{y'[x] == 3*y[x], y[0] == 2}, y[x], x]

- compare NDSolve[{y'[x] == 3*y[x], y[0] == 2}, y, {x,0,2}] along with y[1]/.% and Plot[Evaluate[y[x] /.% line number],{x,0,3}]

- For coupled equations dv/dt = -x and dx/dt = v (SHO with omega = 1)
  NDSolve[{v'[t] == -x[t], x'[t] == v[t], v[0] == 0, x[0] == 1}, {v, x}, {t, 0, 6.28}]

User-Defined Functions:

- name your functions and variables in all lower case
- underscore = “to be assigned later”
- name[a_, b_] := messy expression that’s a function of a and b

- to evaluate: name[2,3]

- to plot: Plot[name[2, z], {z, 0, 4}]
  or Plot3D[name[x, y], {x, 0, 3}, {y, 0, 3}]