Pymol Reference Card

Modes

Pymol supports two modes of input: point and click mode, and command line mode. The point and click allows you to quickly rotate the molecule(s) zoom in and out and change the clipping planes. The command line mode where commands are entered into the external GUI window supports all of the commands in the point and click mode, but is more flexible and possibly useful for complex selection and command issuing. Commands entered on the command line are executed when you press the return key.

command help	`help` *keyword*

Loading Files

file loading	`load data/test/pept.pdb`
loading from terminal	`pymol data/test/pept.pdb`
toggle between text and graphics	*Esc*
toggle Y axis rocking	`rock`
stereo view	`stereo on/off`
stereo type	`stereo crosseye / walleye / quadbuffer`
undo action	`undo`
reset view	`reset`
reinitialize Pymol	`reinitialize`
quit (force, even if unsaved)	`quit`

Mouse Control

	L	M	R	Wheel
	Rota	Move	MovZ	Slab
Shift	+Box	-Box	Clip	MovS
Ctrl	+/-	PkAt	Pk1	—
CtSh	Sele	Cent	Menu	—
DblClk	Menu	Cent	PkAt	—

set the center of rotation `origin` *selection*

Atom Selection

object-name/segi-id/chain-id/resi-id/name-id

molecular system selection	`/pept`
molecule selection	`/pept/lig`
chain selection	`/pept/lig/a`
residue selection	`/pept/lig/a/10`
atom	`/pept/lig/a/10/ca`
ranges	`lig/a/10-12/ca`
ranges	`a/6+8/c+o`
missing selections	`/pept//a`
naming a selection	`select bb, name c+o+n+ca`
count atoms in a selection	`count_atoms bb`
remove atoms from a selection	`remove resi 5`
general	`all, none, hydro, hetatm, visible, present`
atoms not in a selection	`select sidechains, ! bb`
atoms with a vdW gap < 3 Å	`resi 6 around 3`
atom centers with a gap < 1.0 Å	`all near 1 of resi 6`
atom centers within < 4.0 Å	`all within 4 of resi 6`

Basic Commands

Some commands used with atoms selections. If you are unsure about the selection, click on the molecule part that you want in the viewing window and then look at the output line to see the selection.

fill viewer with selection	`zoom /pept//a`
center a selection	`center /pept//a`
colour a selection	`colour pink, /pept//a`
force Pymol to reapply colours	`recolor`
set background colour	`bg_color white`
vdW representation of selection	`show spheres, 156/ca`
stick representation of selection	`show sticks, a//`
line representation of selection	`show lines, /pept`
ribbon representation of selection	`show ribbon, /pept`
dot representation of selection	`show dots, /pept`
mesh representation of selection	`show mesh, /pept`
surface representation of selection	`show surface, /pept`
nonbonded representation of selection	`show nonbonded, /pept`
nonbonded sphere representation of selection	`show nb_spheres, /pept`
cartoon representation of selection	`show cartoon, a//`
clear all	`hide all`
rotate a selection	`rotate` *axis*, *angle*, *selection*
translate a selection	`translate [x,y,z],` *selection*

Cartoon Settings

Setting the value at the end to 0 forces the secondary structure to go though the CA position.

cylindrical helices	`set cartoon_cylindrical_helices,1`
fancy helices [tubular edge]	`set cartoon_fancy_helices,1`
flat sheets	`set cartoon_flat_sheets,1`
smooth loops	`set cartoon_smooth_loops,1`
find rings for cartoon	`set cartoon_ring_finder,[1,2,3,4]`
ring mode	`set cartoon_ring_mode,[1,2,3]`
nucleic acid mode	`set nucleic_acid_mode,[0,1,2,3,4]`
cartoon sidechains	`set cartoon_side_chain_helper; rebuild`
primary colour	`set cartoon_color,blue`
secondary colour	`set cartoon_highlight_color,grey`
limit colour to ss	`set cartoon_discrete_colors,on`
cartoon transparency	`set cartoon_transparency,0.5`
cartoon loop	`cartoon loop, a//`
cartoon loop	`cartoon loop, a//`
cartoon rectangular	`cartoon rect, a//`
cartoon oval	`cartoon oval, a//`
cartoon tubular	`cartoon tube, a//`
cartoon arrow	`cartoon arrow, a//`
cartoon dumbell	`cartoon dumbell, a//`
b-factor sausage	`cartoon putty, a//`

Image Output

low resolution	`ray`
high resolution	`ray 2000,2000`
ultra-high resolution	`ray 5000,5000`
change the default size [pts]	`viewport 640,480`
image shadow control	`set ray_shadow,0`
image fog control	`set ray_trace_fog,0`
image depth cue control	`set depth_cue,0`
image antialiasing control	`set antialias,1`
export image as .png	`png` *image*`.png`

Hydrogen Bonding

Draw bonds between atoms and label the residues that are involved.

draw a line between atoms	`distance 542/oe1,538/ne`
set the line dash gap	`set dash_gap,0.09`
set the line dash width	`set dash_width,3.0`
set the line dash radius	`set dash_radius,0.0`
set the line dash length	`set dash_length,0.15`
set round dash ends	`set dash_round_ends,on`
hide a label	`hide labels, dist01`
label a reside	`label (542/oe1), "%s" %("E542")`
set label font	`set label_font_id,4`
set label colour	`set label_color,white`

Electrostatics

There are a number of ways to apply electrostatics in Pymol. The user can use GRASP to generate a map and then import it. Alternatively the user can use the APBS Pymol plugin. Pymol also has a built in function that is quick and dirty.

generate electrostatic surface `action > generate>vacuum electrostatics > protein contact potential`

Pymol Movies (mac)

move the camera	`move x,10`
turn the camera	`turn x,90`
play the movie	`mplay`
stop the movie	`mstop`
writeout png files	`mpng` *prefix* `[, first [, last]]`
show a particular frame	`frame` *number*
move forward on frame	`forward`
move back one frame	`backwards`
go to the start of the movie	`rewind`
go to the middle of the movie	`middle`
go to the movie end	`ending`
determine the current frame	`get_frame`
clear the movie cache	`mclear`
execute a command in a frame	`mdo 1, turn x,5; turn y,5;`
dump current movie commands	`mdump`
reset the number of frames per second	`meter_reset`

Miscellaneous

add hydrogens in to a molecule selection	`h_add`
alias a set of commands separated by ";"	`alias go,load 1hpv.pdb; zoom 200/; show sticks, 200/ around 8`
structurally align	`align prot1////CA, prot2, object=alignment`
fit one molelcule to another	`fit` *selection, target*
copy at selection	`copy` *target, source*
create a new selection	`create` *target, selection*
delete a selection	`delete` *selection*
save file	`save` *filename, selection*
protect or deprotect a selection	`[de]protect` *selection*
mask or demask to allow/stop selection	`[un]mask` *selection*
align coordinates with axis	`orient` *selection*
get the current rotation matrix	`get_view`
input a rotation matrix	`set_view`
safely refresh the scene	`refresh`
store a scene	`view` *name*, `store,` *description*
restore a view	`view` *name*, `[recall]`
set a new colour	`set_color` *name, rgb*

Secondary Structures

Pymol has a secondary structure determination algorithm called dss, however it is better to use the DSSP algorithm and then define the limits manually.

to run dss	`dss` *selection*
to define helical structure	`alter 11-40/, ss='H'`
to define loop regions	`alter 40-50/, ss='L'`
to define strand structure	`alter 50-60/, ss='S'`
rebuild the cartoon after alteration	`rebuild`
get dihedral angle	`get_dihedral 4/n,4/c,4/ca,4/cb`

Files

change the working directory	`cd <path>`
list contents of current directory	`ls`
print current working directory	`pwd`

Crystal Structures

To recreate crystal packing of molelcules within 5 Å of pept in the pept.pdb (which must contain CRYST date), use the symexp command. `symexp sym,pept,(pept),5.0`

NMR Structures

NMR models should be loaded into the same object, but should have different states. load a model into an object `load` *file.pdb, object*

show all models in an object	`set all_states,1`
show only one object model	`set all_states,0`
show a particular model	`frame` *model_number*
determine which model	`get_model`
fit two structures to one another	`fit` *selection*
fit and calculate the rms	`rms` *selection*
rms without fitting	`rms_cur` *selection*
fit ensemble structures	`intra_fit` *selection*`,1`
calculate rms	`intra_rms` *selection,state*
ensemble rms without fitting	`intra_rms_cur` *selection,state*

Changing Structures

add a bond	`bond atom1, atom2`
remove bonds	`unbond atom1,atom2`
join to molecules together	`fuse [atom1, atom2]`

Old School Images

Load a .pdb and make a cartoon view. Then change the background colour to white and change the ray mode to 2.

	`set ray_trace_mode,2`
make the lines thinner	`set antialias,2`
raytrace the image	`ray`