GMT-3.1 QUICK REFERENCE 1

FILTERING OF 1-D AND 2-D DATA:	
blockmean/median/mode	L_2, L_1, and mode estimate (x,y,z) data filters/decimators
filter1d	Filter 1-D data (time series)
grdfilter	Filter 2-D data in space domain

PLOTTING OF 1-D and 2-D DATA:	
grdcontour	Contouring of 2-D gridded data
grdimage	Produce images from 2-D gridded data
grdvector	Plot vector fields from 2-D gridded data
grdview	3-D perspective imaging of 2-D gridded data
psbasemap	Create a basemap frame
psclip	Use polygon files as clipping paths
pscoast	Plot coastlines, filled continents, rivers, and political borders
pscontour	Direct contouring or imaging of *xyz* data by triangulation
pshistogram	Plot a histogram
psimage	Plot Sun rasterfiles on a map
psmask	Create overlay to mask specified regions of a map
psrose	Plot sector or rose diagrams
psscale	Plot grayscale or colorscale
pstext	Plot textstrings
pswiggle	Draw anomalies along track
psxy	Plot symbols, polygons, and lines in 2-D
psxyz	Plot symbols, polygons, and lines in 3-D

GRIDDING OF (X,Y,Z) TABLE DATA:	
nearneighbor	Nearest-neighbor gridding scheme
surface	Continuous curvature gridding algorithm
triangulate	Perform optimal Delauney triangulation on *xyz* data

SAMPLING OF 1-D AND 2-D DATA:	
grdsample	Resample a 2-D gridded data onto new grid
grdtrack	Sampling of 2-D data along 1-D track
sample1d	Resampling of 1-D data

PROJECTION AND MAP-TRANSFORMATION:	
grdproject	Transform gridded data to a new coordinate system
mapproject	Transform table data to a new coordinate system
project	Project data onto lines or great circles

INFORMATION:	
gmtdefaults	List the current default settings
gmtset	Command-line editing of parameters in the .gmtdefaults file
grdinfo	Get information about the content of gridded files
minmax	Report extreme values in table data files

CONVERT OR EXTRACT SUBSETS OF DATA:	
gmtconvert	Convert table data from one format to another
gmtselect	Select table data subsets based on multiple spatial criteria
grd2xyz	Convert 2-D gridded data to table data
grdcut	Cut a sub-region from a gridded file
grdpaste	Paste together gridded files along common edge
grdreformat	Convert from one grid format to another
splitxyz	Split *xyz* table data into several segments
xyz2grd	Convert table data to 2-D gridded file

MISCELLANEOUS:	
makecpt	Create GMT color palette tables
spectrum1d	Compute spectral estimates from time-series
triangulate	Perform optimal Delauney triangulation on xyz data

DETERMINE TRENDS IN 1-D AND 2-D DATA:	
fitcircle	Finds best-fitting great or small circles
grdtrend	Fits polynomial trends to gridded files ($z = f(x,y)$)
trend1d	Fits polynomial or Fourier trends to $y = f(x)$ series
trend2d	Fits polynomial trends to $z = f(x,y)$ series

OTHER OPERATIONS ON 2-D GRIDS:	
grd2cpt	Make color palette table from gridded file
grdclip	Limit the *z*-range in gridded data sets
grdedit	Modify grid header information
grdfft	Operate on gridded files in frequency domain
grdgradient	Compute directional gradients from gridded files
grdhisteq	Histogram equalization for gridded files
grdlandmask	Creates mask gridded file from coastline database
grdmask	Set grid nodes in/outside a clip path to constants
grdmath	Reverse Polish Notaton (RPN) calculator for gridded files
grdvolume	Calculate volume under a surface within a contour

STANDARDIZED COMMAND LINE OPTIONS:	
–**B***xtick*[*:text:*][*/ytick*[*:text:*]][*/ztick*[*:text:*]][*WESNZwesnz+*][*:.title:*]	Tickmarks
–**H***[n_headers]*	Input/output ascii tables have header record[s]
–**J** Map projection. Give width or use lower case code and specify map scale	
–**JA***lon_0/lat_0/width*	Lambert azimuthal equal area
–**JB***lon_0/lat_0/lat_1/lat_2/width*	Albers conic equal area
–**JC***lon_0/lat_0/width*	Cassini cylindrical
–**JE***lon_0/lat_0/width*	Azimuthal equidistant
–**JF***lon_0/lat_0horizon/width*	Azimuthal Gnomonic
–**JG***lon_0/lat_0/width*	Azimuthal orthographic
–**JH***lon_0/width*	Hammer equal area
–**JI***lon_0/width*	Sinusoidal equal area
–**JJ***lon_0/width*	Miller cylindrical
–**JK***lon_0/width*	Eckert IV equal area
–**JL***lon_0/lat_0/lat_1/lat_2/width*	Lambert conic conformal
–**JM***width* or –**JM***lon_0/lat_0/width*	Mercator cylindrical
–**JN***lon_0/width*	Robinson
–**JOa***lon_0/lat_0/az/width*	Oblique Mercator, 1: origin and azimuth
–**JOb***lon_0/lat_0/lon_1/lat_1/width*	Oblique Mercator, 2: two points
–**JOc***lon_0/lat_0/lon_p/lat_p/width*	Oblique Mercator, 3: origin and pole
–**JP***width*	Polar (θ, r) (or cylindrical)
–**JQ***lon_0/width*	Equidistant cylindrical (Plate Carré)
–**JR***lon_0/width*	Winkel Tripel
–**JS***lon_0/lat_0/width*	General stereographic
–**JT***lon_0/width*	Transverse Mercator
–**JU***zone/width*	Universal Transverse Mercator (UTM)
–**JW***lon_0/width*	Mollweide
–**JX***width*[**l**|**p**][*/height*[**l**|**p**]][**d**]	Linear, $\log_{10}$, and x^a–y^b
–**JY***lon_0/lat_s/width*	General cylindrical equal area
–**K**	Append more PostScript later
–**O**	This is an overlay plot
–**P**	Select Portrait orientation
–**R***west/east/south/north*[*/zmin/zmax*][**r**]	Specify Region of interest
–**U**[*/dx/dy/*][*label*]	Plot time-stamp on plot
–**V**	Run in verbose mode
–**X***off* –**Y***off*	Shift lower left corner of plot [1/1]
–**:**	Expect *y*/*x* input rather than *x*/*y*
–**c***copies*	Set number of plot copies [1]