PostGIS 2.1 Raster Cheatsheet

New in this release [1] Enhanced in this release [2] Requires GEOS 3.3 or higher[g3.3] 2.5/3D support[3d] SQL-MM[mm] Supports geography [G]

Raster Support Data types

geomval   A spatial datatype with two fields - geom (holding a geometry object) and val (holding a double precision pixel value from a raster band).

addbandarg   A composite type used as input into the ST_AddBand function defining the attributes and initial value of the new band.

rastbandarg   A composite type for use when needing to express a raster and a band index of that raster.

raster   raster spatial data type.

reclassarg   A composite type used as input into the ST_Reclass function defining the behavior of reclassification.

unionarg   A composite type used as input into the ST_Union function defining the bands to be processed and behavior of the UNION operation.

Raster Management

AddRasterConstraints   Adds raster constraints to a loaded raster table for a specific column that constrains spatial ref, scaling, blocksize, alignment, bands, band type and a flag to denote if raster column is regularly blocked. The table must be loaded with data for the constraints to be inferred. Returns true of the constraint setting was accomplished and if issues a notice.

```
1. rasttable, rastcolumn, srid, scale_x, scale_y, blocksize_x, blocksize_y, same_alignment, regular_blocking,
   num_bands=true, pixel_types=true, nodata_values=true, out_db=true, extent=true
2. rasttable, rastcolumn, VARIADIC constraints
3. rastschema, rasttable, rastcolumn, VARIADIC constraints
4. rastschema, rasttable, rastcolumn, srid=true, scale_x=true, scale_y=true, blocksize_x=true, blocksize_y=true,
   same_alignment=true, regular_blocking=false, num_bands=true, pixel_types=true, nodata_values=true, out_db=true,
   extent=true
```

DropRasterConstraints   Drops PostGIS raster constraints that refer to a raster table column. Useful if you need to reload data or update your raster column data.

```
1. rasttable, rastcolumn, srid, scale_x, scale_y, blocksize_x, blocksize_y, same_alignment, regular_blocking,
   num_bands=true, pixel_types=true, nodata_values=true, out_db=true, extent=true
2. rastschema, rasttable, rastcolumn, srid=true, scale_x=true, scale_y=true, blocksize_x=true, blocksize_y=true,
   same_alignment=true, regular_blocking=false, num_bands=true, pixel_types=true, nodata_values=true, out_db=true,
   extent=true
3. rastschema, rasttable, rastcolumn, constraints
```

PostGIS_Raster_Lib_Build_Date ()   Reports full raster library build date.

PostGIS_Raster_Lib_Version ()   Reports full raster version and build configuration infos.

ST_GDALDrivers ()   Returns a list of raster formats supported by your lib gdal. These are the formats you can output your raster using ST_AsGDALRaster.

UpdateRasterSRID[1]   Change the SRID of all rasters in the user-specified column and table.

```
1. schema_name, table_name, column_name, new_srid
2. table_name, column_name, new_srid
```

Raster Constructors

ST_AddBand[2]   Returns a raster with the new band(s) of given type added with given initial value in the given index location. If no index is specified, the band is added to the end.

```
1. rast, addbandargset
2. rast, pixeltype, initialvalue=0, nodataval=NULL
3. rast, index, pixeltype, initialvalue=0, nodataval=NULL
4. torast, fromrast, fromband=1, torastindex=at_end
5. torast, fromrasts, fromband=1, torastindex=at_end
```

ST_AsRaster   Converts a PostGIS geometry to a PostGIS raster.

```
 1. geom, ref, pixeltype, value=1, nodataval=0, touched=false
 2. geom, ref, pixeltype=ARRAY['8BUI'], value=ARRAY[1], nodataval=ARRAY[0], touched=false
 3. geom, scalex, scaley, gridx, gridy, pixeltype, value=1, nodataval=0, skewx=0, skewy=0, touched=false
 4. geom, scalex, scaley, gridx=NULL, gridy=NULL, pixeltype=ARRAY['8BUI'], value=ARRAY[1], nodataval=ARRAY[0], skewx=0,
    skewy=0, touched=false
 5. geom, scalex, scaley, pixeltype, value=1, nodataval=0, upperleftx=NULL, upperlefty=NULL, skewx=0, skewy=0,
    touched=false
 6. geom, scalex, scaley, pixeltype, value=ARRAY[1], nodataval=ARRAY[0], upperleftx=NULL, upperlefty=NULL, skewx=0,
    skewy=0, touched=false
 7. geom, width, height, gridx, gridy, pixeltype, value=1, nodataval=0, skewx=0, skewy=0, touched=false
 8. geom, width, height, gridx=NULL, gridy=NULL, pixeltype=ARRAY['8BUI'], value=ARRAY[1], nodataval=ARRAY[0], skewx=0,
    skewy=0, touched=false
 9. geom, width, height, pixeltype, value=1, nodataval=0, upperleftx=NULL, upperlefty=NULL, skewx=0, skewy=0, touched=false
10. geom, width, height, pixeltype, value=ARRAY[1], nodataval=ARRAY[0], upperleftx=NULL, upperlefty=NULL, skewx=0, skewy=0,
    touched=false
```

ST_Band   Returns one or more bands of an existing raster as a new raster. Useful for building new rasters from existing rasters.

```
1. rast, nbands = ARRAY[1]
2. rast, nbands, delimiter=,
3. rast, nband
```

ST_MakeEmptyRaster   Returns an empty raster (having no bands) of given dimensions (width & height), upperleft X and Y, pixel size and rotation (scalex, scaley, skewx & skewy) and reference system (srid). If a raster is passed in, returns a new raster with the same size, alignment and SRID. If srid is left out, the spatial ref is set to unknown (0).

```
1. rast
2. width, height, upperleftx, upperlefty, scalex, scaley, skewx, skewy, srid=unknown
3. width, height, upperleftx, upperlefty, pixelsize
```

ST_Tile[1]   Returns a set of rasters resulting from the split of the input raster based upon the desired dimensions of the output rasters.

```
1. rast, nband, width, height, padwithnodata=FALSE, nodataval=NULL
2. rast, nband, width, height, padwithnodata=FALSE, nodataval=NULL
3. rast, width, height, padwithnodata=FALSE, nodataval=NULL
```

Raster Accessors
ST_GeoReference (rast, format=GDAL)   Returns the georeference meta data in GDAL or ESRI format as commonly seen in a world file. Default is GDAL.
ST_Height (rast)   Returns the height of the raster in pixels.
ST_MetaData (rast)   Returns basic meta data about a raster object such as pixel size, rotation (skew), upper, lower left, etc.
ST_NumBands (rast)   Returns the number of bands in the raster object.
ST_PixelHeight (rast)   Returns the pixel height in geometric units of the spatial reference system.
ST_PixelWidth (rast)   Returns the pixel width in geometric units of the spatial reference system.
ST_ScaleX (rast)   Returns the X component of the pixel width in units of coordinate reference system.
ST_ScaleY (rast)   Returns the Y component of the pixel height in units of coordinate reference system.
ST_RasterToWorldCoord[1] (rast, xcolumn, yrow)   Returns the raster's upper left corner as geometric X and Y (longitude and latitude) given a column and row. Column and row starts at 1.

ST_RasterToWorldCoordX   Returns the geometric X coordinate upper left of a raster, column and row. Numbering of columns and rows starts at 1.
 	
    1. rast, xcolumn
    2. rast, xcolumn, yrow

ST_RasterToWorldCoordY   Returns the geometric Y coordinate upper left corner of a raster, column and row. Numbering of columns and rows starts at 1.
 	
    1. rast, yrow
    2. rast, xcolumn, yrow

ST_Rotation (rast)   Returns the rotation of the raster in radian.
ST_SkewX (rast)   Returns the georeference X skew (or rotation parameter).
ST_SkewY (rast)   Returns the georeference Y skew (or rotation parameter).
ST_SRID (rast)   Returns the spatial reference identifier of the raster as defined in spatial_ref_sys table.
ST_UpperLeftX (rast)   Returns the upper left X coordinate of raster in projected spatial ref.
ST_UpperLeftY (rast)   Returns the upper left Y coordinate of raster in projected spatial ref.
ST_Width (rast)   Returns the width of the raster in pixels.

ST_WorldToRasterCoord[1]   Returns the upper left corner as column and row given geometric X and Y (longitude and latitude) or a point geometry expressed in the spatial reference coordinate system of the raster.
 	
    1. rast, pt
    2. rast, longitude, latitude

ST_WorldToRasterCoordX   Returns the column in the raster of the point geometry (pt) or a X and Y world coordinate (xw, yw) represented in world spatial reference system of raster.
 	
    1. rast, pt
    2. rast, xw
    3. rast, xw, yw

ST_WorldToRasterCoordY   Returns the row in the raster of the point geometry (pt) or a X and Y world coordinate (xw, yw) represented in world spatial reference system of raster.
 	
    1. rast, pt
    2. rast, xw
    3. rast, xw, yw

ST_IsEmpty (rast)   Returns true if the raster is empty (width = 0 and height = 0). Otherwise, returns false.

Raster Band Accessors
ST_BandMetaData (rast, bandnum=1)   Returns basic meta data for a specific raster band. band num 1 is assumed if none-specified.
ST_BandNoDataValue (rast, bandnum=1)   Returns the value in a given band that represents no data. If no band num 1 is assumed.

ST_BandIsNoData   Returns true if the band is filled with only nodata values.
 	
    1. rast, band, forceChecking=true
    2. rast, forceChecking=true

ST_BandPath (rast, bandnum=1)   Returns system file path to a band stored in file system. If no bandnum specified, 1 is assumed.
ST_BandPixelType (rast, bandnum=1)   Returns the type of pixel for given band. If no bandnum specified, 1 is assumed.
ST_HasNoBand (rast, bandnum=1)   Returns true if there is no band with given band number. If no band number is specified, then band number 1 is assumed.

Raster Pixel Accessors and Setters
ST_PixelAsPolygon (rast, columnx, rowy)   Returns the polygon geometry that bounds the pixel for a particular row and column.
ST_PixelAsPolygons[2] (rast, band=1, exclude_nodata_value=TRUE)   Returns the polygon geometry that bounds every pixel of a raster band along with the value, the X and the Y raster coordinates of each pixel.
ST_PixelAsPoint[1] (rast, columnx, rowy)   Returns a point geometry of the pixel's upper-left corner.
ST_PixelAsPoints[1] (rast, band=1, exclude_nodata_value=TRUE)   Returns a point geometry for each pixel of a raster band along with the value, the X and the Y raster coordinates of each pixel. The coordinates of the point geometry are of the pixel's upper-left corner.
ST_PixelAsCentroid[1] (rast, columnx, rowy)   Returns the centroid (point geometry) of the area represented by a pixel.
ST_PixelAsCentroids[1] (rast, band=1, exclude_nodata_value=TRUE)   Returns the centroid (point geometry) for each pixel of a raster band along with the value, the X and the Y raster coordinates of each pixel. The point geometry is the centroid of the area represented by a pixel.

ST_Value   Returns the value of a given band in a given columnx, rowy pixel or at a particular geometric point. Band numbers start at 1 and assumed to be 1 if not specified. If exclude_nodata_value is set to false, then all pixels include nodata pixels are considered to intersect and return value. If exclude_nodata_value is not passed in then reads it from metadata of raster.
 	
    1. rast, pt, exclude_nodata_value=true
    2. rast, bandnum, pt, exclude_nodata_value=true
    3. rast, columnx, rowy, exclude_nodata_value=true
    4. rast, bandnum, columnx, rowy, exclude_nodata_value=true

ST_NearestValue[1]   Returns the nearest non-NODATA value of a given band's pixel specified by a columnx and rowy or a geometric point expressed in the same spatial reference coordinate system as the raster.

```
1. rast, bandnum, pt, exclude_nodata_value=true
2. rast, pt, exclude_nodata_value=true
3. rast, bandnum, columnx, rowy, exclude_nodata_value=true
4. rast, columnx, rowy, exclude_nodata_value=true
```

ST_Neighborhood[1]   Returns a 2-D double precision array of the non-NODATA values around a given band's pixel specified by either a columnX and rowY or a geometric point expressed in the same spatial reference coordinate system as the raster.

```
1. rast, bandnum, columnX, rowY, distanceX, distanceY, exclude_nodata_value=true
2. rast, columnX, rowY, distanceX, distanceY, exclude_nodata_value=true
3. rast, bandnum, pt, distanceX, distanceY, exclude_nodata_value=true
4. rast, pt, distanceX, distanceY, exclude_nodata_value=true
```

ST_SetValue[2]   Returns modified raster resulting from setting the value of a given band in a given columnx, rowy pixel or the pixels that intersect a particular geometry. Band numbers start at 1 and assumed to be 1 if not specified.

```
1. rast, bandnum, geom, newvalue
2. rast, geom, newvalue
3. rast, bandnum, columnx, rowy, newvalue
4. rast, columnx, rowy, newvalue
```

ST_SetValues[1]   Returns modified raster resulting from setting the values of a given band.

```
1. rast, nband, columnx, rowy, newvalueset, noset=NULL, keepnodata=FALSE
2. rast, nband, columnx, rowy, newvalueset, nosetvalue, keepnodata=FALSE
3. rast, nband, columnx, rowy, width, height, newvalue, keepnodata=FALSE
4. rast, columnx, rowy, width, height, newvalue, keepnodata=FALSE
5. rast, nband, geomvalset, keepnodata=FALSE
```

ST_DumpValues[1]   Get the values of the specified band as a 2-dimension array.

```
1. rast, nband, exclude_nodata_value=true
2. rast, nband, exclude_nodata_value=true
```

ST_PixelOfValue[1]   Get the columnx, rowy coordinates of the pixel whose value equals the search value.

```
1. rast, nband, search, exclude_nodata_value=true
2. rast, search, exclude_nodata_value=true
3. rast, nband, search, exclude_nodata_value=true
4. rast, search, exclude_nodata_value=true
```

Raster Editors

ST_SetGeoReference (rast, georefcoords, format=GDAL)   Set Georeference 6 georeference parameters in a single call. Numbers should be separated by white space. Accepts inputs in GDAL or ESRI format. Default is GDAL.

ST_SetRotation (rast, rotation)   Set the rotation of the raster in radian.

ST_SetScale   Sets the X and Y size of pixels in units of coordinate reference system. Number units/pixel width/height.

```
1. rast, xy
2. rast, x, y
```

ST_SetSkew   Sets the georeference X and Y skew (or rotation parameter). If only one is passed in, sets X and Y to the same value.

```
1. rast, skewxy
2. rast, skewx, skewy
```

ST_SetSRID (rast, srid)   Sets the SRID of a raster to a particular integer srid defined in the spatial_ref_sys table.

ST_SetUpperLeft (rast, x, y)   Sets the value of the upper left corner of the pixel to projected X and Y coordinates.

ST_Resample   Resample a raster using a specified resampling algorithm, new dimensions, an arbitrary grid corner and a set of raster georeferencing attributes defined or borrowed from another raster.

```
1. rast, width, height, gridx=NULL, gridy=NULL, skewx=0, skewy=0, algorithm=NearestNeighbour, maxerr=0.125
2. rast, scalex=0, scaley=0, gridx=NULL, gridy=NULL, skewx=0, skewy=0, algorithm=NearestNeighbor, maxerr=0.125
3. rast, ref, algorithm=NearestNeighbour, maxerr=0.125, usescale=true
4. rast, ref, usescale, algorithm=NearestNeighbour, maxerr=0.125
```

ST_Rescale   Resample a raster by adjusting only its scale (or pixel size). New pixel values are computed using the NearestNeighbor (english or american spelling), Bilinear, Cubic, CubicSpline or Lanczos resampling algorithm. Default is NearestNeighbor.

```
1. rast, scalexy, algorithm=NearestNeighbour, maxerr=0.125
2. rast, scalex, scaley, algorithm=NearestNeighbour, maxerr=0.125
```

ST_Reskew   Resample a raster by adjusting only its skew (or rotation parameters). New pixel values are computed using the NearestNeighbor (english or american spelling), Bilinear, Cubic, CubicSpline or Lanczos resampling algorithm. Default is NearestNeighbor.

```
1. rast, skewxy, algorithm=NearestNeighbour, maxerr=0.125
2. rast, skewx, skewy, algorithm=NearestNeighbour, maxerr=0.125
```

ST_SnapToGrid   Resample a raster by snapping it to a grid. New pixel values are computed using the NearestNeighbor (english or american spelling), Bilinear, Cubic, CubicSpline or Lanczos resampling algorithm. Default is NearestNeighbor.

```
1. rast, gridx, gridy, algorithm=NearestNeighbour, maxerr=0.125, scalex=DEFAULT 0, scaley=DEFAULT 0
2. rast, gridx, gridy, scalex, scaley, algorithm=NearestNeighbour, maxerr=0.125
3. rast, gridx, gridy, scalexy, algorithm=NearestNeighbour, maxerr=0.125
```

ST_Resize[1]   Resize a raster to a new width/height

```
1. rast, width, height, algorithm=NearestNeighbor, maxerr=0.125
2. rast, percentwidth, percentheight, algorithm=NearestNeighbor, maxerr=0.125
3. rast, width, height, algorithm=NearestNeighbor, maxerr=0.125
```

Raster Band Editors

ST_SetBandNoDataValue   Sets the value for the given band that represents no data. Band 1 is assumed if no band is specified. To mark a band as having no nodata value, set the nodata value = NULL.

```
1. rast, nodatavalue
2. rast, band, nodatavalue, forcechecking=false
```

ST_SetBandIsNoData (rast, band=1)   Sets the isnodata flag of the band to TRUE.

Raster Band Statistics and Analytics

ST_Count   Returns the number of pixels in a given band of a raster or raster coverage. If no band is specified defaults to band 1. If exclude_nodata_value is set to true, will only count pixels that are not equal to the nodata value.

```
1. rast, nband=1, exclude_nodata_value=true
2. rast, exclude_nodata_value
3. rastertable, rastercolumn, nband=1, exclude_nodata_value=true
4. rastertable, rastercolumn, exclude_nodata_value
```

ST_Histogram   Returns a set of record summarizing a raster or raster coverage data distribution separate bin ranges. Number of bins are autocomputed if not specified.

```
1. rast, nband=1, exclude_nodata_value=true, bins=autocomputed, width=NULL, right=false
2. rast, nband, bins, width=NULL, right=false
3. rast, nband, exclude_nodata_value, bins, right
4. rast, nband, bins, right
5. rastertable, rastercolumn, nband, bins, right
6. rastertable, rastercolumn, nband, exclude_nodata_value, bins, right
7. rastertable, rastercolumn, nband=1, exclude_nodata_value=true, bins=autocomputed, width=NULL, right=false
8. rastertable, rastercolumn, nband=1, bins, width=NULL, right=false
```

ST_Quantile   Compute quantiles for a raster or raster table coverage in the context of the sample or population. Thus, a value could be examined to be at the raster's 25%, 50%, 75% percentile.

```
 1. rast, nband=1, exclude_nodata_value=true, quantiles=NULL
 2. rast, quantiles
 3. rast, nband, quantiles
 4. rast, quantile
 5. rast, exclude_nodata_value, quantile=NULL
 6. rast, nband, quantile
 7. rast, nband, exclude_nodata_value, quantile
 8. rast, nband, quantile
 9. rastertable, rastercolumn, nband=1, exclude_nodata_value=true, quantiles=NULL
10. rastertable, rastercolumn, nband, quantiles
```

ST_SummaryStats   Returns record consisting of count, sum, mean, stddev, min, max for a given raster band of a raster or raster coverage. Band 1 is assumed is no band is specified.

```
1. rastertable, rastercolumn, exclude_nodata_value
2. rast, exclude_nodata_value
3. rastertable, rastercolumn, nband=1, exclude_nodata_value=true
4. rast, nband, exclude_nodata_value
```

ST_ValueCount   Returns a set of records containing a pixel band value and count of the number of pixels in a given band of a raster (or a raster coverage) that have a given set of values. If no band is specified defaults to band 1. By default nodata value pixels are not counted. and all other values in the pixel are output and pixel band values are rounded to the nearest integer.

```
 1. rast, nband=1, exclude_nodata_value=true, searchvalues=NULL, roundto=0,
 2. rast, nband, searchvalues, roundto=0,
 3. rast, searchvalues, roundto=0,
 4. rast, searchvalue, roundto=0
 5. rast, nband, exclude_nodata_value, searchvalue, roundto=0
 6. rast, nband, searchvalue, roundto=0
 7. rastertable, rastercolumn, nband=1, exclude_nodata_value=true, searchvalues=NULL, roundto=0,
 8. rastertable, rastercolumn, searchvalues, roundto=0,
 9. rastertable, rastercolumn, nband, searchvalues, roundto=0,
10. rastertable, rastercolumn, nband, exclude_nodata_value, searchvalue, roundto=0
11. rastertable, rastercolumn, searchvalue, roundto=0
12. rastertable, rastercolumn, nband, searchvalue, roundto=0
```

Raster Outputs

ST_AsBinary (rast)   Return the Well-Known Binary (WKB) representation of the raster without SRID meta data.

ST_AsGDALRaster (rast, format, options=NULL, srid=sameassource)   Return the raster tile in the designated GDAL Raster format. Raster formats are one of those supported by your compiled library. Use ST_GDALRasters() to get a list of formats supported by your library.

ST_AsJPEG   Return the raster tile selected bands as a single Joint Photographic Exports Group (JPEG) image (byte array). If no band is specified and 1 or more than 3 bands, then only the first band is used. If only 3 bands then all 3 bands are used and mapped to RGB.

```
1. rast, options=NULL
2. rast, nband, quality
3. rast, nband, options=NULL
4. rast, nbands, options=NULL
5. rast, nbands, quality
```

ST_AsPNG   Return the raster tile selected bands as a single portable network graphics (PNG) image (byte array). If 1, 3, or 4 bands in raster and no bands are specified, then all bands are used. If more 2 or more than 4 bands and no bands specified, then only band 1 is used. Bands are mapped to RGB or RGBA space.

```
1. rast, options=NULL
2. rast, nband, compression
3. rast, nband, options=NULL
4. rast, nbands, compression
5. rast, nbands, options=NULL
```

ST_AsTIFF   Return the raster selected bands as a single TIFF image (byte array). If no band is specified, then will try to use all bands.

```
1. rast, options='', srid=sameassource
2. rast, compression='', srid=sameassource
```

```
    3. rast, nbands, compression='', srid=sameassource
    4. rast, nbands, options, srid=sameassource
```

Raster Processing

Box3D (rast)   Returns the box 3d representation of the enclosing box of the raster.

ST_Clip[2]   Returns the raster clipped by the input geometry. If band number not is specified, all bands are processed. If crop is not specified or TRUE, the output raster is cropped.

```
    1. rast, nband, geom, nodataval=NULL, crop=TRUE
    2. rast, nband, geom, nodataval, crop=TRUE
    3. rast, nband, geom, crop
    4. rast, geom, nodataval=NULL, crop=TRUE
    5. rast, geom, nodataval, crop=TRUE
    6. rast, geom, crop
```

ST_ConvexHull (rast)   Return the convex hull geometry of the raster including pixel values equal to BandNoDataValue. For regular shaped and non-skewed rasters, this gives the same result as ST_Envelope so only useful for irregularly shaped or skewed rasters.

ST_DumpAsPolygons (rast, band_num=1, exclude_nodata_value=TRUE)   Returns a set of geomval (geom,val) rows, from a given raster band. If no band number is specified, band num defaults to 1.

ST_Envelope (rast)   Returns the polygon representation of the extent of the raster.

ST_HillShade[2]   Returns the hypothetical illumination of an elevation raster band using provided azimuth, altitude, brightness and scale inputs.

```
    1. rast, band=1, pixeltype=32BF, azimuth=315, altitude=45, max_bright=255, scale=1.0, interpolate_nodata=FALSE
    2. rast, band, customextent, pixeltype=32BF, azimuth=315, altitude=45, max_bright=255, scale=1.0, interpolate_nodata=FALSE
```

ST_Aspect[2]   Returns the aspect (in degrees by default) of an elevation raster band. Useful for analyzing terrain.

```
    1. rast, band=1, pixeltype=32BF, units=DEGREES, interpolate_nodata=FALSE
    2. rast, band, customextent, pixeltype=32BF, units=DEGREES, interpolate_nodata=FALSE
```

ST_Slope[2]   Returns the slope (in degrees by default) of an elevation raster band. Useful for analyzing terrain.

```
    1. rast, nband=1, pixeltype=32BF, units=DEGREES, scale=1.0, interpolate_nodata=FALSE
    2. rast, nband, customextent, pixeltype=32BF, units=DEGREES, scale=1.0, interpolate_nodata=FALSE
```

ST_Intersection   Returns a raster or a set of geometry-pixelvalue pairs representing the shared portion of two rasters or the geometrical intersection of a vectorization of the raster and a geometry.

```
    1. geom, rast, band_num=1
    2. rast, geom
    3. rast, band_num, geom
    4. rast1, rast2, nodataval
    5. rast1, rast2, returnband='BOTH', nodataval=NULL
    6. rast1, band_num1, rast2, band_num2, nodataval
    7. rast1, band_num1, rast2, band_num2, returnband='BOTH', nodataval=NULL
```

ST_MapAlgebra[1]   Callback function version - Returns a one-band raster given one or more input rasters, band indexes and one user-specified callback function.

```
    1. rastbandargset, callbackfunc, pixeltype=NULL, extenttype=INTERSECTION, customextent=NULL, distancex=0, distancey=0,
       VARIADIC userargs=NULL
    2. rast, nband, callbackfunc, pixeltype=NULL, extenttype=FIRST, customextent=NULL, distancex=0, distancey=0, VARIADIC
       userargs=NULL
    3. rast, nband, callbackfunc, pixeltype=NULL, extenttype=FIRST, customextent=NULL, distancex=0, distancey=0, VARIADIC
       userargs=NULL
    4. rast1, nband1, rast2, nband2, callbackfunc, pixeltype=NULL, extenttype=INTERSECTION, customextent=NULL, distancex=0,
       distancey=0, VARIADIC userargs=NULL
```

ST_MapAlgebra[1]   Expression version - Returns a one-band raster given one or two input rasters, band indexes and one or more user-specified SQL expressions.

```
    1. rast, nband, pixeltype, expression, nodataval=NULL
    2. rast, pixeltype, expression, nodataval=NULL
    3. rast1, nband1, rast2, nband2, expression, pixeltype=NULL, extenttype=INTERSECTION, nodata1expr=NULL, nodata2expr=NULL,
       nodatanodataval=NULL
    4. rast1, rast2, expression, pixeltype=NULL, extenttype=INTERSECTION, nodata1expr=NULL, nodata2expr=NULL,
       nodatanodataval=NULL
```

ST_MapAlgebraExpr   1 raster band version: Creates a new one band raster formed by applying a valid PostgreSQL algebraic operation on the input raster band and of pixeltype provided. Band 1 is assumed if no band is specified.

```
    1. rast, band, pixeltype, expression, nodataval=NULL
    2. rast, pixeltype, expression, nodataval=NULL
```

ST_MapAlgebraExpr   2 raster band version: Creates a new one band raster formed by applying a valid PostgreSQL algebraic operation on the two input raster bands and of pixeltype provided. band 1 of each raster is assumed if no band numbers are specified. The resulting raster will be aligned (scale, skew and pixel corners) on the grid defined by the first raster and have its extent defined by the "extenttype" parameter. Values for "extenttype" can be: INTERSECTION, UNION, FIRST, SECOND.

```
    1. rast1, rast2, expression, pixeltype=same_as_rast1_band, extenttype=INTERSECTION, nodata1expr=NULL, nodata2expr=NULL,
       nodatanodataval=NULL
    2. rast1, band1, rast2, band2, expression, pixeltype=same_as_rast1_band, extenttype=INTERSECTION, nodata1expr=NULL,
       nodata2expr=NULL, nodatanodataval=NULL
```

ST_MapAlgebraFct   1 band version - Creates a new one band raster formed by applying a valid PostgreSQL function on the input raster band and of pixeltype prodived. Band 1 is assumed if no band is specified.

```
    1. rast, onerasteruserfunc
    2. rast, onerasteruserfunc, VARIADIC args
    3. rast, pixeltype, onerasteruserfunc
    4. rast, pixeltype, onerasteruserfunc, VARIADIC args
    5. rast, band, onerasteruserfunc
    6. rast, band, onerasteruserfunc, VARIADIC args
    7. rast, band, pixeltype, onerasteruserfunc
    8. rast, band, pixeltype, onerasteruserfunc, VARIADIC args
```

ST_MapAlgebraFct   2 band version - Creates a new one band raster formed by applying a valid PostgreSQL function on the 2 input raster bands and of pixeltype prodived. Band 1 is assumed if no band is specified. Extent type defaults to INTERSECTION if not specified.

```
1. rast1, rast2, tworastuserfunc, pixeltype=same_as_rast1, extenttype=INTERSECTION, VARIADIC userargs
2. rast1, band1, rast2, band2, tworastuserfunc, pixeltype=same_as_rast1, extenttype=INTERSECTION, VARIADIC userargs
```

ST_MapAlgebraFctNgb (rast, band, pixeltype, ngbwidth, ngbheight, onerastngbuserfunc, nodatamode, VARIADIC args)   1-band version: Map Algebra Nearest Neighbor using user-defined PostgreSQL function. Return a raster which values are the result of a PLPGSQL user function involving a neighborhood of values from the input raster band.

ST_Polygon[2] (rast, band_num=1)   Returns a multipolygon geometry formed by the union of pixels that have a pixel value that is not no data value. If no band number is specified, band num defaults to 1.

ST_Reclass   Creates a new raster composed of band types reclassified from original. The nband is the band to be changed. If nband is not specified assumed to be 1. All other bands are returned unchanged. Use case: convert a 16BUI band to a 8BUI and so forth for simpler rendering as viewable formats.

```
1. rast, nband, reclassexpr, pixeltype, nodataval=NULL
2. rast, VARIADIC reclassargset
3. rast, reclassexpr, pixeltype
```

ST_Union[1][2]   Returns the union of a set of raster tiles into a single raster composed of 1 or more bands.

```
1. rast
2. rast, unionargset
3. rast, nband
4. rast, uniontype
5. rast, nband, uniontype
```

Raster Processing Builtin Functions

ST_Min4ma[2]   Raster processing function that calculates the minimum pixel value in a neighborhood.

```
1. matrix, nodatamode, VARIADIC args
2. value, pos, VARIADIC userargs
```

ST_Max4ma[2]   Raster processing function that calculates the maximum pixel value in a neighborhood.

```
1. matrix, nodatamode, VARIADIC args
2. value, pos, VARIADIC userargs
```

ST_Sum4ma[2]   Raster processing function that calculates the sum of all pixel values in a neighborhood.

```
1. matrix, nodatamode, VARIADIC args
2. value, pos, VARIADIC userargs
```

ST_Mean4ma[2]   Raster processing function that calculates the mean pixel value in a neighborhood.

```
1. matrix, nodatamode, VARIADIC args
2. value, pos, VARIADIC userargs
```

ST_Range4ma[2]   Raster processing function that calculates the range of pixel values in a neighborhood.

```
1. matrix, nodatamode, VARIADIC args
2. value, pos, VARIADIC userargs
```

ST_Distinct4ma[2]   Raster processing function that calculates the number of unique pixel values in a neighborhood.

```
1. matrix, nodatamode, VARIADIC args
2. value, pos, VARIADIC userargs
```

ST_StdDev4ma[2]   Raster processing function that calculates the standard deviation of pixel values in a neighborhood.

```
1. matrix, nodatamode, VARIADIC args
2. value, pos, VARIADIC userargs
```

ST_InvDistWeight4ma[1] (value, pos, VARIADIC userargs)   Raster processing function that interpolates a pixel's value from the pixel's neighborhood.

ST_MinDist4ma[1] (value, pos, VARIADIC userargs)   Raster processing function that returns the minimum distance (in number of pixels) between the pixel of interest and a neighboring pixel with value.

Raster Operators

&& (A, B)   Returns TRUE if A's bounding box intersects B's bounding box.

&< (A, B)   Returns TRUE if A's bounding box is to the left of B's.

&> (A, B)   Returns TRUE if A's bounding box is to the right of B's.

Raster and Raster Band Spatial Relationships

ST_Contains[1]   Return true if no points of raster rastB lie in the exterior of raster rastA and at least one point of the interior of rastB lies in the interior of rastA.

```
1. rastA, nbandA, rastB, nbandB
2. rastA, rastB
```

ST_ContainsProperly[1]   Return true if rastB intersects the interior of rastA but not the boundary or exterior of rastA.

```
1. rastA, nbandA, rastB, nbandB
2. rastA, rastB
```

ST_Covers[1]   Return true if no points of raster rastB lie outside raster rastA.

```
1. rastA, nbandA, rastB, nbandB
2. rastA, rastB
```

ST_CoveredBy[1]   Return true if no points of raster rastA lie outside raster rastB.

```
1. rastA, nbandA, rastB, nbandB
2. rastA, rastB
```

ST_Disjoint[1]   Return true if raster rastA does not spatially intersect rastB.

```
1. rastA, nbandA, rastB, nbandB
2. rastA, rastB
```

ST_Intersects   Return true if raster rastA spatially intersects raster rastB.

```
1. rastA, nbandA, rastB, nbandB
2. rastA, rastB
3. rast, nband, geommin
4. rast, geommin, nband=NULL
5. geommin, rast, nband=NULL
```

ST_Overlaps[1]   Return true if raster rastA and rastB intersect but one does not completely contain the other.

```
1. rastA, nbandA, rastB, nbandB
2. rastA, rastB
```

ST_Touches[1]   Return true if raster rastA and rastB have at least one point in common but their interiors do not intersect.

```
1. rastA, nbandA, rastB, nbandB
2. rastA, rastB
```

ST_SameAlignment[2]   Returns true if rasters have same skew, scale, spatial ref and false if they don't with notice detailing issue.

```
1. rastA, rastB
2. ulx1, uly1, scalex1, scaley1, skewx1, skewy1, ulx2, uly2, scalex2, scaley2, skewx2, skewy2
3. rastfield
```

ST_NotSameAlignmentReason[1] (rastA, rastB)   Returns text stating if rasters are aligned and if not aligned, a reason why.

ST_Within[1]   Return true if no points of raster rastA lie in the exterior of raster rastB and at least one point of the interior of rastA lies in the interior of rastB.

```
1. rastA, nbandA, rastB, nbandB
2. rastA, rastB
```

ST_DWithin[1]   Return true if rasters rastA and rastB are within the specified distance of each other.

```
1. rastA, nbandA, rastB, nbandB, distance_of_srid
2. rastA, rastB, distance_of_srid
```

ST_DFullyWithin[1]   Return true if rasters rastA and rastB are fully within the specified distance of each other.

```
1. rastA, nbandA, rastB, nbandB, distance_of_srid
2. rastA, rastB, distance_of_srid
```