pyeastronomy library

# LLD

LLD.lat holds latitudinal coordinate in radians

LLD.lon holds longitudinal coordinate in radians

LLD.dst holds distance coordinate in AU or km depending on the function

LLD(lat, lon, dst) creates an LLD object

LLD.str() returns a string representation (lat, lon, dst) in native units

LLD.str\_EC() returns a string representation (lat, lon, dst) with lat & lon in decimal degrees, as typically used for Ecliptic coordinates

LLD.str\_EQ() returns a string representation (lat, lon, dst) with lat in decimal degrees and lon in decimal hour angle, as typically used for Equatorial coordinates

LLD == LLD returns true if all 3 coordinates are equal

LLD += LLD adds all 3 coordinates individually

LLD \*= double multiplies lat & lon coordinates by the given value

# vec3, vec4 & dvec4

object.x holds x coordinate, unit depends on function

object.y holds y coordinate, unit depends on function

object.z holds z coordinate, unit depends on function

object.w holds w coordinate for vec4 and dvec4, unit depends on function

dvec4 uses double precision (type double), whereas vec3 & vec4 uses single precision (type float)

# EDateTime

EDateTime() creates the object at current UTC system time

EDateTime(year, month, day, hour, minute, second) creates the object at provided UTC time. year & month are signed integers; day, hour, minute, second are double’s

EDateTime(jd, is\_tt) creates the object at provided jd (Julian Day) taken to be UTC, unless the optional bool is\_tt is set to true (it is false by default)

EDateTime(unix\_timestamp) creates the object at the given UNIX timestamp (long integer)

EDateTime.year() returns the year

EDateTime.month() returns the month

EDateTime.day() returns the day

EDateTime.hour() returns the hour

EDateTime.minute() returns the minute

EDateTime.second() returns the (fractional) second

EDateTime.jd\_tt() returns the Julian Day in Terrestrial (Dynamic) time

EDateTime.jd\_utc() returns the Julian Day in UTC time (used for sidereal time calculations)

EDateTime.setTimeNow() sets the time and date to the current system time, converted to UTC