MeteoLib Functions

MeteoLib is located in the com.raytheon.edex.meteoLib package.

The wrapper file, Controller.java, includes documentation to explain most of the functions in the library and what parameters are taken in and what values are returned. Some components return floats, others return float arrays, while others return objects containing multiple returns. The user will have to then use "get" functions to access the data contained within these.

To access meteoLib functions the calls must be of this fashion:
Controller.*function_name*(*parameter1*,*parameter2*,...);

For multiple returns that the user must create an object:
Object *objectname* = new Object();
objectname.*getObjectVarName*();

The Controller.java file is the file that communicates with the other wrapped files allowing the user to send in items in a manner that they would like to send them in.

The following are the objects that can be called and what variables they contain to return. These functions serve no purpose but to simply hold and return variables. First are the variables and their types followed by the functions used to access them.

Index.java

Variables

float totalIndex
float crossTotalIndex
float verticalTotalsIndex

Functions

getTotalIndex()
getCrossTotalIndex()
getVerticalTotalsIndex()

Motion.java

Variables

float direction
float speed
float uComp
float vComp

Functions

getDirection()
getSpeed()
getUComp()
getVComp()

PHT.java

Variables

- float temperature
- float temperature1
- float pressure
- float pressure1
- float dewpoint
- float height
- float height1
- float wetBulbTemp
- float positiveEnergy
- float cin;
- float dryAdiabat
- float moistAdiabat
- float[] heightArray
- float[] pressureArray
- float[] temperatureArray
- float[] dewpointArray
- float[] virtualTemps
- float[] soundingTemps
- float[] soundingVirtTemps
- float mixingRatio
- int numLevels
- int completion
- int status

Functions

- getTemperature()
- getTemperature1 ()
- getPressure()
- getPressure1 ()
- getDewpoint ()
- getHeight()
- getHeight1 ()
- getWetBulbTemp ()
- getPositiveEnergy ()
- getCin()
- getDryAdiabat ()
- getMoistAdiabat()
- getHeightArray ()
- getPressureArray ()
- getTemperatureArray ()
- getDewpointArray()
- getVirtualTemps()
- getSoundingTemps ()
- getSoundingVirtTemps ()

getMixingRatio()
getNumLevels()
getCompletion ()
getStatus()

Tsoar.java

Variables

float potentialTempForecast
float heightMinimumEffectiveConvection
float tempMinimumEffectiveConvection
float heightMaxThermalAltssssssss
float tempMaxThermalAlt
float soarIndex
float triggerTemperature

Functions

getPotentialTempForecast()
getHeightMinimumEffectiveConvection ()
getTempMinimumEffectiveConvection ()
getHeightMaxThermalAlt()
getTempMaxThermalAlt ()
getSoarIndex()
getTriggerTemperature()

VectorVars.java

Variables

float[] qx
float[] qy
float[] slqy
float[] slqx
float[] dadxdt
float[] dadydt
float[] bx
float[] by
float minimum
float maximum
float range

Functions

getQx ()
getQy()
getSlqy ()
getSlqx ()
getDadxdt ()
getDadydt ()
getBx ()

getBy()
getMinimum ()
getMaximum ()
getRange ()

Velocity.java

Variables

float[] verticalVelocity
float maxVerticalVelocity

Functions

getVerticalVelocity()
getMaxVerticalVelocity()

WindComp.java

Variables

float windDirection
float windSpeed
float uComp
float vComp
float[] uCompArray
float[] vCompArray
float[] windDirectionArray
float[] windSpeedArray
float stormMotionDir
float stormMotionSpd
float stormRelativeHelicity
float helicity
float[] compFirstInSecond
float[] compFirstInKDir
int gustPotential

Functions

getWindDirection ()
getWindSpeed ()
getUComp ()
getVComp()
getUCompArray ()
getVCompArray()
getWindDirectionArray ()
getWindSpeedArray ()
getStormMotionDir ()
getStormMotionSpd()
getStormRelativeHelicity ()
getHelicity()
getCompFirstInSecond ()

getCompFirstInKDir()
getGustPotential()