GeoSAMS GUI

Thomas Callaghan

Version 0.1

Contents

[GeoSAMS GUI i](#_Toc188506514)

[Thomas Callaghan i](#_Toc188506515)

[Version 0.1 i](#_Toc188506516)

[1 GeoSAMS GUI 1](#_Toc188506517)

[1.1 SHOW Args 1](#_Toc188506518)

[1.2 START Sim 1](#_Toc188506519)

[1.3 SAVE ALL Configs 1](#_Toc188506520)

[1.4 Year in file names 1](#_Toc188506521)

[2 Math Setup Frame 3](#_Toc188506522)

[2.1 Radio Button 3](#_Toc188506523)

[2.2 Write Startup File 3](#_Toc188506524)

[3 Set Fishing Mortality in Special Access Areas 4](#_Toc188506525)

[3.1 Number Defined 4](#_Toc188506526)

[3.2 Load and Save Fishing Mortality Files 4](#_Toc188506527)

[3.3 Area SubFrames 4](#_Toc188506528)

[3.3.1 Year Definitions 4](#_Toc188506529)

[3.3.2 Corners, or Fields of Defined Mortalities by Area 4](#_Toc188506530)

[4 Growth Frame 5](#_Toc188506531)

[4.1 Mortality 5](#_Toc188506532)

[4.1.1 Fishing Mortality 5](#_Toc188506533)

[4.1.2 Alpha Mortality 5](#_Toc188506534)

[4.1.3 Adult Mortality 5](#_Toc188506535)

[4.1.4 Computing Alpha 5](#_Toc188506536)

[4.1.5 Computing Natural Mortality 5](#_Toc188506537)

[4.2 Selectivity 6](#_Toc188506538)

[4.3 Incidental 6](#_Toc188506539)

[4.4 Discard 6](#_Toc188506540)

[4.5 Overall Mortality, M 6](#_Toc188506541)

[4.6 Computing Landings Per Unit Effort, LPUE 6](#_Toc188506542)

[5 Main 8](#_Toc188506543)

[5.1 Growth subframe that identifies 8](#_Toc188506544)

[5.2 Recuitment 8](#_Toc188506545)

[5.3 Configuration Files 8](#_Toc188506546)

[5.4 Output Selection 8](#_Toc188506547)

[6 Sort By Area Frame 9](#_Toc188506548)

[6.1 Number of Areas 9](#_Toc188506549)

[6.2 Output Parameters 9](#_Toc188506550)

[6.3 Load and Save Data Sort Files 9](#_Toc188506551)

[6.4 Run Sort 9](#_Toc188506552)

[6.5 Area SubFrames 9](#_Toc188506553)

[6.5.1 YYYY 9](#_Toc188506554)

[6.5.2 Comment 9](#_Toc188506555)

[6.5.3 # Corners 9](#_Toc188506556)

[6.5.4 Corner N 9](#_Toc188506557)

[7 Sort By Region Frame 11](#_Toc188506558)

[7.1 Output Parameters 11](#_Toc188506559)

[7.2 Run Sort 11](#_Toc188506560)

[8 Special Access Area 12](#_Toc188506561)

[8.1 Number of Areas 12](#_Toc188506562)

[8.2 Special Access File 12](#_Toc188506563)

[8.3 Area Definitions 12](#_Toc188506564)

[8.3.1 Area N 12](#_Toc188506565)

[8.3.2 Corners: 12](#_Toc188506566)

[8.3.3 Corner N 12](#_Toc188506567)

[9 Namespace Index 13](#_Toc188506568)

[9.1 Package List 13](#_Toc188506569)

[10 Hierarchical Index 14](#_Toc188506570)

[10.1 Class Hierarchy 14](#_Toc188506571)

[11 Class Index 16](#_Toc188506572)

[11.1 Class List 16](#_Toc188506573)

[12 File Index 17](#_Toc188506574)

[12.1 File List 17](#_Toc188506575)

[13 Namespace Documentation 18](#_Toc188506576)

[13.1 AreaManager Namespace Reference 18](#_Toc188506577)

[13.1.1 Classes 18](#_Toc188506578)

[13.2 EditMathSetupFrame Namespace Reference 19](#_Toc188506579)

[13.2.1 Classes 19](#_Toc188506580)

[13.3 FishMortBySpecAcc Namespace Reference 20](#_Toc188506581)

[13.3.1 Classes 20](#_Toc188506582)

[13.4 GeoSams Namespace Reference 21](#_Toc188506583)

[13.4.1 Classes 21](#_Toc188506584)

[13.4.2 Functions 21](#_Toc188506585)

[13.4.3 Function Documentation 21](#_Toc188506586)

[13.5 Globals Namespace Reference 22](#_Toc188506587)

[13.5.1 Functions 22](#_Toc188506588)

[13.5.2 Variables 22](#_Toc188506589)

[13.5.3 Function Documentation 22](#_Toc188506590)

[13.5.4 Variable Documentation 24](#_Toc188506591)

[13.6 GrowthFrame Namespace Reference 26](#_Toc188506592)

[13.6.1 Classes 26](#_Toc188506593)

[13.7 MainInputFrame Namespace Reference 27](#_Toc188506594)

[13.7.1 Classes 27](#_Toc188506595)

[13.8 PointInPolygon Namespace Reference 28](#_Toc188506596)

[13.8.1 Functions 28](#_Toc188506597)

[13.8.2 Function Documentation 28](#_Toc188506598)

[13.9 shapefile Namespace Reference 29](#_Toc188506599)

[13.9.1 Classes 29](#_Toc188506600)

[13.9.2 Functions 29](#_Toc188506601)

[13.9.3 Variables 29](#_Toc188506602)

[13.9.4 Detailed Description 30](#_Toc188506603)

[13.9.5 Function Documentation 30](#_Toc188506604)

[13.9.6 Variable Documentation 32](#_Toc188506605)

[13.10 ShapeTest Namespace Reference 34](#_Toc188506606)

[13.10.1 Classes 34](#_Toc188506607)

[13.10.2 class GeoShapeVariables 34](#_Toc188506608)

[13.10.3 Variable Documentation 35](#_Toc188506609)

[13.11 SortByAreaFrame Namespace Reference 36](#_Toc188506610)

[13.11.1 Classes 36](#_Toc188506611)

[13.12 SortByRegionFrame Namespace Reference 37](#_Toc188506612)

[13.12.1 Classes 37](#_Toc188506613)

[13.13 SortIntoColumns Namespace Reference 38](#_Toc188506614)

[13.13.1 Classes 38](#_Toc188506615)

[13.13.2 Variables 38](#_Toc188506616)

[13.13.3 Variable Documentation 39](#_Toc188506617)

[13.14 SpecialAreaFrame Namespace Reference 41](#_Toc188506618)

[13.14.1 Classes 41](#_Toc188506619)

[13.15 Widgets Namespace Reference 42](#_Toc188506620)

[13.15.1 Classes 42](#_Toc188506621)

[13.15.2 Functions 42](#_Toc188506622)

[13.15.3 Function Documentation 42](#_Toc188506623)

[14 Class Documentation 43](#_Toc188506624)

[14.1 shapefile.\_Array Class Reference 43](#_Toc188506625)

[14.1.1 Public Member Functions 43](#_Toc188506626)

[14.1.2 Detailed Description 43](#_Toc188506627)

[14.1.3 Member Function Documentation 43](#_Toc188506628)

[14.2 shapefile.\_Record Class Reference 44](#_Toc188506629)

[14.2.1 Public Member Functions 44](#_Toc188506630)

[14.2.2 Private Attributes 44](#_Toc188506631)

[14.2.3 Detailed Description 45](#_Toc188506632)

[14.2.4 Constructor & Destructor Documentation 45](#_Toc188506633)

[14.2.5 Member Function Documentation 45](#_Toc188506634)

[14.2.6 Member Data Documentation 46](#_Toc188506635)

[14.3 AreaManager.AreaManager Class Reference 47](#_Toc188506636)

[14.3.1 Public Member Functions 47](#_Toc188506637)

[14.3.2 Public Attributes 47](#_Toc188506638)

[14.3.3 Detailed Description 48](#_Toc188506639)

[14.3.4 Constructor & Destructor Documentation 48](#_Toc188506640)

[14.3.5 Member Function Documentation 48](#_Toc188506641)

[14.3.6 Member Data Documentation 49](#_Toc188506642)

[14.4 AreaManager.AreaMgrSubFrame Class Reference 50](#_Toc188506643)

[14.4.1 Public Member Functions 50](#_Toc188506644)

[14.4.2 Public Attributes 51](#_Toc188506645)

[14.4.3 Constructor & Destructor Documentation 51](#_Toc188506646)

[14.4.4 Member Function Documentation 51](#_Toc188506647)

[14.4.5 Member Data Documentation 52](#_Toc188506648)

[14.5 SortIntoColumns.Column Class Reference 53](#_Toc188506649)

[14.5.1 Public Member Functions 53](#_Toc188506650)

[14.5.2 Public Attributes 53](#_Toc188506651)

[14.5.3 Constructor & Destructor Documentation 53](#_Toc188506652)

[14.5.4 Member Data Documentation 53](#_Toc188506653)

[14.6 AreaManager.Corner Class Reference 54](#_Toc188506654)

[14.6.1 Public Member Functions 54](#_Toc188506655)

[14.6.2 Public Attributes 54](#_Toc188506656)

[14.6.3 Detailed Description 54](#_Toc188506657)

[14.6.4 Constructor & Destructor Documentation 54](#_Toc188506658)

[14.6.5 Member Data Documentation 54](#_Toc188506659)

[14.7 EditMathSetupFrame.EditMathSetup Class Reference 55](#_Toc188506660)

[14.7.1 Public Member Functions 55](#_Toc188506661)

[14.7.2 Public Attributes 56](#_Toc188506662)

[14.7.3 Detailed Description 56](#_Toc188506663)

[14.7.4 Constructor & Destructor Documentation 56](#_Toc188506664)

[14.7.5 Member Function Documentation 56](#_Toc188506665)

[14.7.6 Member Data Documentation 57](#_Toc188506666)

[14.8 FishMortBySpecAcc.FishMortBySpecAcc Class Reference 58](#_Toc188506667)

[14.8.1 Public Member Functions 58](#_Toc188506668)

[14.8.2 Public Attributes 59](#_Toc188506669)

[14.8.3 Detailed Description 59](#_Toc188506670)

[14.8.4 Constructor & Destructor Documentation 59](#_Toc188506671)

[14.8.5 Member Function Documentation 60](#_Toc188506672)

[14.8.6 Member Data Documentation 61](#_Toc188506673)

[14.9 ShapeTest.GeoShape Class Reference 62](#_Toc188506674)

[14.9.1 Public Member Functions 62](#_Toc188506675)

[14.9.2 Public Attributes 62](#_Toc188506676)

[14.9.3 Constructor & Destructor Documentation 62](#_Toc188506677)

[14.9.4 Member Data Documentation 62](#_Toc188506678)

[14.10 SortByRegionFrame.GeoShape Class Reference 63](#_Toc188506679)

[14.10.1 Public Member Functions 63](#_Toc188506680)

[14.10.2 Public Attributes 63](#_Toc188506681)

[14.10.3 Detailed Description 63](#_Toc188506682)

[14.10.4 Constructor & Destructor Documentation 63](#_Toc188506683)

[14.10.5 Member Data Documentation 63](#_Toc188506684)

[14.11 SortIntoColumns.GeoShape Class Reference 64](#_Toc188506685)

[14.11.1 Public Member Functions 64](#_Toc188506686)

[14.11.2 Public Attributes 64](#_Toc188506687)

[14.11.3 Constructor & Destructor Documentation 64](#_Toc188506688)

[14.11.4 Member Data Documentation 64](#_Toc188506689)

[14.12 GrowthFrame.Growth Class Reference 65](#_Toc188506690)

[14.12.1 Public Member Functions 67](#_Toc188506691)

[14.12.2 Public Attributes 68](#_Toc188506692)

[14.12.3 Detailed Description 70](#_Toc188506693)

[14.12.4 Constructor & Destructor Documentation 70](#_Toc188506694)

[14.12.5 Member Function Documentation 70](#_Toc188506695)

[14.12.6 Member Data Documentation 71](#_Toc188506696)

[14.13 GeoSams.MainApplication Class Reference 74](#_Toc188506697)

[14.13.1 Public Member Functions 76](#_Toc188506698)

[14.13.2 Public Attributes 77](#_Toc188506699)

[14.13.3 Detailed Description 77](#_Toc188506700)

[14.13.4 Constructor & Destructor Documentation 77](#_Toc188506701)

[14.13.5 Member Function Documentation 78](#_Toc188506702)

[14.13.6 Member Data Documentation 81](#_Toc188506703)

[14.14 MainInputFrame.MainInput Class Reference 83](#_Toc188506704)

[14.14.1 Public Member Functions 84](#_Toc188506705)

[14.14.2 Public Attributes 85](#_Toc188506706)

[14.14.3 Detailed Description 86](#_Toc188506707)

[14.14.4 Constructor & Destructor Documentation 87](#_Toc188506708)

[14.14.5 Member Function Documentation 87](#_Toc188506709)

[14.14.6 Member Data Documentation 89](#_Toc188506710)

[14.15 shapefile.Reader Class Reference 92](#_Toc188506711)

[14.15.1 Public Member Functions 94](#_Toc188506712)

[14.15.2 Public Attributes 94](#_Toc188506713)

[14.15.3 Protected Attributes 94](#_Toc188506714)

[14.15.4 Private Member Functions 94](#_Toc188506715)

[14.15.5 Private Attributes 95](#_Toc188506716)

[14.15.6 Detailed Description 95](#_Toc188506717)

[14.15.7 Constructor & Destructor Documentation 95](#_Toc188506718)

[14.15.8 Member Function Documentation 95](#_Toc188506719)

[14.15.9 Member Data Documentation 100](#_Toc188506720)

[14.16 Widgets.ScrollFrame Class Reference 102](#_Toc188506721)

[14.16.1 Public Member Functions 102](#_Toc188506722)

[14.16.2 Public Attributes 103](#_Toc188506723)

[14.16.3 Detailed Description 103](#_Toc188506724)

[14.16.4 Constructor & Destructor Documentation 103](#_Toc188506725)

[14.16.5 Member Function Documentation 103](#_Toc188506726)

[14.16.6 Member Data Documentation 104](#_Toc188506727)

[14.17 shapefile.Shape Class Reference 105](#_Toc188506728)

[14.17.1 Public Member Functions 105](#_Toc188506729)

[14.17.2 Public Attributes 105](#_Toc188506730)

[14.17.3 Static Protected Member Functions 105](#_Toc188506731)

[14.17.4 Protected Attributes 106](#_Toc188506732)

[14.17.5 \_errorsPrivate Attributes 106](#_Toc188506733)

[14.17.6 Constructor & Destructor Documentation 106](#_Toc188506734)

[14.17.7 Member Function Documentation 106](#_Toc188506735)

[14.17.8 Member Data Documentation 107](#_Toc188506736)

[14.18 shapefile.ShapefileException Class Reference 108](#_Toc188506737)

[14.18.1 Detailed Description 108](#_Toc188506738)

[14.19 shapefile.ShapeRecord Class Reference 109](#_Toc188506739)

[14.19.1 Public Member Functions 109](#_Toc188506740)

[14.19.2 Public Attributes 109](#_Toc188506741)

[14.19.3 Detailed Description 109](#_Toc188506742)

[14.19.4 Constructor & Destructor Documentation 109](#_Toc188506743)

[14.19.5 Member Function Documentation 109](#_Toc188506744)

[14.19.6 Member Data Documentation 110](#_Toc188506745)

[14.20 shapefile.ShapeRecords Class Reference 111](#_Toc188506746)

[14.20.1 Public Member Functions 111](#_Toc188506747)

[14.20.2 Detailed Description 111](#_Toc188506748)

[14.20.3 Member Function Documentation 111](#_Toc188506749)

[14.21 shapefile.Shapes Class Reference 112](#_Toc188506750)

[14.21.1 Public Member Functions 112](#_Toc188506751)

[14.21.2 Detailed Description 112](#_Toc188506752)

[14.21.3 Member Function Documentation 112](#_Toc188506753)

[14.22 SortByAreaFrame.SortByArea Class Reference 113](#_Toc188506754)

[14.22.1 Public Member Functions 115](#_Toc188506755)

[14.22.2 Public Attributes 115](#_Toc188506756)

[14.22.3 Detailed Description 116](#_Toc188506757)

[14.22.4 Constructor & Destructor Documentation 116](#_Toc188506758)

[14.22.5 Member Function Documentation 116](#_Toc188506759)

[14.22.6 Member Data Documentation 118](#_Toc188506760)

[14.23 SortByRegionFrame.SortByRegion Class Reference 120](#_Toc188506761)

[14.23.1 Public Member Functions 121](#_Toc188506762)

[14.23.2 Public Attributes 121](#_Toc188506763)

[14.23.3 Detailed Description 122](#_Toc188506764)

[14.23.4 Constructor & Destructor Documentation 122](#_Toc188506765)

[14.23.5 Member Function Documentation 122](#_Toc188506766)

[14.23.6 Member Data Documentation 124](#_Toc188506767)

[14.24 SpecialAreaFrame.SpecialArea Class Reference 126](#_Toc188506768)

[14.24.1 Public Member Functions 127](#_Toc188506769)

[14.24.2 Public Attributes 127](#_Toc188506770)

[14.24.3 Detailed Description 127](#_Toc188506771)

[14.24.4 Constructor & Destructor Documentation 127](#_Toc188506772)

[14.24.5 Member Function Documentation 128](#_Toc188506773)

[14.24.6 Member Data Documentation 129](#_Toc188506774)

[14.25 Widgets.SubFrameElement Class Reference 130](#_Toc188506775)

[14.25.1 Public Member Functions 130](#_Toc188506776)

[14.25.2 Public Attributes 130](#_Toc188506777)

[14.25.3 Detailed Description 130](#_Toc188506778)

[14.25.4 Constructor & Destructor Documentation 130](#_Toc188506779)

[14.25.5 Member Data Documentation 131](#_Toc188506780)

[14.26 Widgets.SubFrameInterpFunction Class Reference 132](#_Toc188506781)

[14.26.1 Public Member Functions 132](#_Toc188506782)

[14.26.2 Public Attributes 132](#_Toc188506783)

[14.26.3 Constructor & Destructor Documentation 133](#_Toc188506784)

[14.26.4 Member Data Documentation 133](#_Toc188506785)

[14.27 Widgets.SubFrameXY Class Reference 134](#_Toc188506786)

[14.27.1 Public Member Functions 134](#_Toc188506787)

[14.27.2 Public Attributes 134](#_Toc188506788)

[14.27.3 Detailed Description 134](#_Toc188506789)

[14.27.4 Constructor & Destructor Documentation 134](#_Toc188506790)

[14.27.5 Member Data Documentation 135](#_Toc188506791)

[14.28 shapefile.Writer Class Reference 136](#_Toc188506792)

[14.28.1 Public Member Functions 138](#_Toc188506793)

[14.28.2 Public Attributes 138](#_Toc188506794)

[14.28.3 Protected Member Functions 138](#_Toc188506795)

[14.28.4 Protected Attributes 138](#_Toc188506796)

[14.28.5 Private Member Functions 139](#_Toc188506797)

[14.28.6 Detailed Description 139](#_Toc188506798)

[14.28.7 Constructor & Destructor Documentation 139](#_Toc188506799)

[14.28.8 Member Function Documentation 139](#_Toc188506800)

[14.28.9 Member Data Documentation 144](#_Toc188506801)

[15 File Documentation 145](#_Toc188506802)

[15.1 AreaManager.py File Reference 145](#_Toc188506803)

[15.1.1 Classes 145](#_Toc188506804)

[15.1.2 class AreaManager.AreaMgrSubFrameNamespaces 145](#_Toc188506805)

[15.2 EditMathSetupFrame.py File Reference 146](#_Toc188506806)

[15.2.1 Classes 146](#_Toc188506807)

[15.2.2 Namespaces 146](#_Toc188506808)

[15.3 FishMortBySpecAcc.py File Reference 147](#_Toc188506809)

[15.3.1 Classes 147](#_Toc188506810)

[15.3.2 Namespaces 147](#_Toc188506811)

[15.4 GeoSams.py File Reference 148](#_Toc188506812)

[15.4.1 Classes 148](#_Toc188506813)

[15.4.2 Namespaces 148](#_Toc188506814)

[15.4.3 Functions 148](#_Toc188506815)

[15.5 Globals.py File Reference 149](#_Toc188506816)

[15.5.1 Namespaces 149](#_Toc188506817)

[15.5.2 Functions 149](#_Toc188506818)

[15.5.3 Variables 149](#_Toc188506819)

[15.6 GrowthFrame.py File Reference 150](#_Toc188506820)

[15.6.1 Classes 150](#_Toc188506821)

[15.6.2 Namespaces 150](#_Toc188506822)

[15.7 MainInputFrame.py File Reference 151](#_Toc188506823)

[15.7.1 Classes 151](#_Toc188506824)

[15.7.2 Namespaces 151](#_Toc188506825)

[15.8 PointInPolygon.py File Reference 152](#_Toc188506826)

[15.8.1 Namespaces 152](#_Toc188506827)

[15.8.2 Functions 152](#_Toc188506828)

[15.9 PyshpMaster/shapefile.py File Reference 153](#_Toc188506829)

[15.9.1 Classes 153](#_Toc188506830)

[15.9.2 Namespaces 153](#_Toc188506831)

[15.9.3 Functions 153](#_Toc188506832)

[15.9.4 Variables 153](#_Toc188506833)

[15.10 ShapeTest.py File Reference 155](#_Toc188506834)

[15.10.1 Classes 155](#_Toc188506835)

[15.10.2 class ShapeTest.GeoShapeNamespaces 155](#_Toc188506836)

[15.10.3 Variables 155](#_Toc188506837)

[15.11 SortByAreaFrame.py File Reference 156](#_Toc188506838)

[15.11.1 Classes 156](#_Toc188506839)

[15.11.2 Namespaces 156](#_Toc188506840)

[15.12 SortByRegionFrame.py File Reference 157](#_Toc188506841)

[15.12.1 Classes 157](#_Toc188506842)

[15.12.2 Namespaces 157](#_Toc188506843)

[15.13 SortIntoColumns.py File Reference 158](#_Toc188506844)

[15.13.1 Classes 158](#_Toc188506845)

[15.13.2 Namespaces 158](#_Toc188506846)

[15.13.3 Variables 158](#_Toc188506847)

[15.14 SpecialAreaFrame.py File Reference 159](#_Toc188506848)

[15.14.1 Classes 159](#_Toc188506849)

[15.14.2 Namespaces 159](#_Toc188506850)

[15.15 Widgets.py File Reference 160](#_Toc188506851)

[15.15.1 Classes 160](#_Toc188506852)

[15.15.2 Namespaces 160](#_Toc188506853)

[15.15.3 Functions 160](#_Toc188506854)

[16 Index 161](#_Toc188506855)

# GeoSAMS GUI

This is the main program for the GeoSAMS GUI

The GUI has 7 tabs

1. **Math Setup** : This frame allows the user to modify the Matlab/Octave startup files.

2. **Main** : Data concerning simulation duration, configutation files in use, and recruitment period

3. **Growth** : Define parameters to compute fishing mortality

4. **Special Access** : Files used to define special areas for fishing management

5. **Fishing Mort in Special Access** : This frame in conjunction with the Special Access Frame is used to define fishing mortalities within a defined area for a specified year. If a location falls within the defined area given by the area defintions in Special Access Frame and assigned the area number.

6. **Sort By Area** : Parameters that are used to sort output data and associate with areas of interest

7. **Sort By Region** : Parameters that are used to sort output data and associate with zones defined by shapefiles

The program is started by entering the following command in the root directory of the workspace $ python .\PythonScripts\GUI\GeoSAM\**GeoSams.py** [10 8]

Where the last two number are optional and used to set limits on:

The maximum number of areas of interest that can be defined.

The maximum number of nodes used to specify each area of interest.

When commanded without these values the GUI defaults to 50 and 8. These values can be viewed by clicking the **SHOW Args** Button

## SHOW Args

As already mention this button is used to show the setup parameters that the GUI is using for maximum number of areas, nodes, and years

## START Sim

This button will start both the GeoSAMS sim and if succsessful continue with the UK interpolation. It does so by first saving the data contained in the other tabs of the GUI to configuraton files specified on this page. It will overwrite the files named if they already exist.

NOTE: The file names listed are part of the package installed when downloaded from GitHub. The user may change these names to preserve the original files. Or reinstall from GitHub to restore the original data.

## SAVE ALL Configs

This is the first step in ***START Sim*** . This button will save all of the configuration files using the names given.

## Year in file names

GUI specifies 2022 to 2025

X\_Y\_BIOM\_2022\_DN Initial state as of June 1, 2022 @ 00:00, i.e. May 31, 2022 @ 24:00

X\_Y\_BIOM\_2023\_DN Growth state as of May 31, 2023 @ 24:00, results for 1st year growth

X\_Y\_BIOM\_2024\_DN Growth state as of May 31, 2024 @ 24:00, results for 2nd year growth

X\_Y\_BIOM\_2025\_DN Growth state as of May 31, 2025 @ 24:00, results for 3rd year growth

X\_Y\_BIOM\_2026\_DN Growth state as of May 31, 2026 @ 24:00, results for 4th year growth

# Math Setup Frame

Matlab should not need any modification as these are the installed directories.

The user should not need to run any Matlab scripts as these are called from the GUI.

Octave on the other hand does require some setup. The user will need to install the desired packages from https://gnu-octave.github.io/packages/

* io
* geometry
* mapping
* statistics

## Radio Button

The user then needs to modify .octaverc to point to where these are installed. Click the Octave radio button and edit the text box to reflect your environment. Then click Write Startup File.

## Write Startup File

When ready, click Write Startup File to save the content to the appropriate startup file

startup.m

.octaverc

Note 1: on a Unix/MacOS platform it is assumed that Matlab is not installed. The script Unpack.sh renames startup.m to startup.xxx so it is not taken for .octaverc

Note 2: If data intialization files have not yet been created, Upack.sh is called to do so.

# Set Fishing Mortality in Special Access Areas

Assists the user in defining areas of interest to assess accumulated parameters located in these areas of interest.

## Number Defined

The number of defined areas as determined by the user. This is limited by Max Areas of Interest. See SHOW Args button for current values.

The Number Defined is limited by default to 25. See SHOW Args for current values. The user can modify this on the command line:

.\PythonScripts\GUI\GeoSAM\**GeoSams.py** Areas Nodes

Default:

.\PythonScripts\GUI\GeoSAM\**GeoSams.py** 25 8

## Load and Save Fishing Mortality Files

The name of the file used to hold this information. The user can load the default file 'FishingMortality.csv' or save their own configuration.

If this feature is not desired then enter NONE in the window

Use Load Fishing Mort File to load a predefined set of data

Use Save Fishing Mort File to save the currently displayed setting

## Area SubFrames

Comment: Optional. Enter a comment to describe the area being specfied.

### Year Definitions

The year for which Area N is valid

### Corners, or Fields of Defined Mortalities by Area

#### Number of Corners

Corners: Specifically, the number of Fields for the year given. This is limited by Max Nodes in Area. See SHOW Args for current values. This can be changed on the command line. See above

#### Field Identifier by Special Area and Mortality

These are the area numbers as determined in Special Access Frame. Enter the area number and its Mortality.

# Growth Frame

Allows the user to modify parameters that are used to define mortality computations.

## Mortality

### Fishing Mortality

This is the default fishing mortality in lieu of any other definition

### Alpha Mortality

So for open areas, an overall fishing mortality would be specified and then at each location would be computed so that:

1. The weighted average (by exploitable numbers), F, over all locations is equal to and

2. at each location is proportional to .

This would also apply to special access areas, but each one would have their own specified F, and the average would only be for those points within that access area.

### Adult Mortality

**Table 1 Mortality**

|  |  |  |
| --- | --- | --- |
|  | **Adult** | **Length0** |
| **MA** | **0.25** | **65.0** |
| **GB** | **0.2** | **70.0** |

### Computing Alpha

Alpha is based on the lengths of the shell normalized by length0

### Computing Natural Mortality

Then natural mortality is computed from juvenile natural mortality and adult natural mortality as

## Selectivity

These parameters are used to compute the scallop selectivity as a function of its length. MA and GB have respective values for each term. GB will also distinguish between open and closed areas.

**Table 2 Selectivity**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **MA** | **GB Open** | **GB Closed** |
| **FSelect A** | **20.5079** | **17.72** | **21.7345** |
| **FSelect B** | **0.19845** | **0.15795** | **0.2193** |

## Incidental

**Table 3 Incidental**

|  |  |
| --- | --- |
| **MA** | **0.05** |
| **GB** | **0.1** |

## Discard

Discard determines how many scallops are thrown out of a catch. It is determined by scallop length and if the area is closed.

if ((length > cull\_size) OR is\_closed) then

SetDiscard = 0.0

else

SetDiscard = discard \* selectivity

**Table 4 Discard**

|  |  |  |
| --- | --- | --- |
|  | **Cull Size** | **Discard** |
| **MA** | **90.0** | **0.2** |
| **GB** | **100.0** | **0.2** |

## Overall Mortality, M

## Computing Landings Per Unit Effort, LPUE

The simulation uses the following parameters to compute LPUE

**Table 5 LPUE**

|  |  |
| --- | --- |
|  | **Default** |
| **LPUE Slope** | **0.6556** |
| **LPUE Slope2** | **2.3** |
| **LPUE Intercept** | **1094.0** |
| **Max # of Scallops Shucked Per Day** | **56000.0** |
| **Max # of Hours Dredging Per Day** | **19.0** |
| **Dredge Width in meters** | **9.144** |
| **Towing Speed in knots** | **4.8** |

# Main

## Growth subframe that identifies

These are the parameters used to control how long the scallop growth is simulated as well as the granularity of the growth computations

Start Year of the simulation

Stop Year of the simulation

Time Steps per year

Domain Name or region of interest, Mid-Atlantic, MA, or Georges Bannk, GB

Sort By Statum: Used when processing Georges-Bank to break the region into quadrants due to its unique shape

## Recuitment

Recruitment is only applied at a certain time of the year. These values determine this period. Combo boxes are used to format the formatting of the month and day.

Start Day, calendar day of the year when recruitment influence begins.

Stop Day, calendar day of the year when reruitment influence ends

## Configuration Files

Files used by the sim to set up parameters. The GUI can use the default values or change the names before starting the sim. The initial names are the default names of the files when first downloaded from GitHub. The names can be changed and the GUI sets up the simulation to use the new names.

## Output Selection

Checkboxes to allow the user to select the desired parameters of interest. This is used to save processing time rather than processing everything. Especially true during interpolation as it would take over and hour to do the interpolation. For example for MA with 11631 grid locations.

Approx 2 minutes per interpolation

Given 3 years worth of data, plus initial conditions

9 listed outputs

Thus 9 x 4 x 2 or 72 minutes. GB is proportionately shorter with only 6802 grid locations.

# Sort By Area Frame

Assists the user in defining areas of interest to assess accumulated parameters located in these areas of interest.

## Number of Areas

The number of defined areas. This is limited by Max Areas of Interest. See SHOW Args button The # of Areas is limited by default to 25. See SHOW Args for current values. The user can modify this on the command line:

.\PythonScripts\GUI\GeoSAM\**GeoSams.py** Areas Nodes

Default (same as started with no arguments):

.\PythonScripts\GUI\GeoSAM\**GeoSams.py** 25 8

## Output Parameters

This is a dropbox of the selected output parameters on the main tab. After a simulation and interpolation have been run, the user would select one of these output, click Run Sort, and the amount of that output in each of the defined areas is accumulated by year to the left of each area.

## Load and Save Data Sort Files

These buttons allow the user to load a predefined set of areas or to save the current set to the named file.

## Run Sort

This will start the program to check if a region grid value for a given year is within one of the specified area and if so accumulate the year sum with that value.

## Area SubFrames

### YYYY

For each year, from Start Year to Stop Year as given in the Main tab an entry box is provided to store the accumulated parameter for that year. These are not populated until after the Run Sort button has been clicked.

### Comment

Optional. Enter a comment to describe the area being specfied.

### # Corners

Also called nodes or sides. This is limited by Max Nodes in Area. See SHOW Args for current values. This can be changed on the command line. See above

### Corner N

These are the coordinates of the area vertices. Enter the Longitude and Latitude of the vertices for the area. It is up to the user to ensure that a closed shape is defined.

# Sort By Region Frame

Assists the user in viewing accumulated parameters located in zones defined by shapefiles.

## Output Parameters

This is a dropbox of the selected output parameters on the main tab. After a simulation and interpolation have been run, the user would select one of these output, click Run Sort, and the amount of that output in each of the shapefile regions is accumulated by year.

## Run Sort

This will start the program to check if a region grid value for a given year is within one of the specified area and if so accumulate the year sum with that value.

# Special Access Area

This frame in conjunction with the FishingMort in Special Access frame is used to define fishing mortalities within a defined area for a specified year.

If a sim data point falls within a defined area given in this frame by the assigned area number. Then if the current year is the same as the year given in the FishingMort in Special Access frame and the area number is listed then the fishing mortality is specified by the Mortality value. Otherwise it is the default value which is defined in the Growth Frame as Fishing mortality #

## Number of Areas

The number of areas the user wishes to define. This is limited by Max Areas of Interest. See SHOW Args button

The # of Areas is limited by default to 25. See SHOW Args. The user can modify this on the command line:

.\PythonScripts\GUI\GeoSAM\**GeoSams.py** #Areas #Nodes Default: python .\PythonScripts\GUI\GeoSAM\**GeoSams.py** 25 8

## Special Access File

The name of the file used to hold this information. The user can load the default file 'SpecialAreas.csv' or define and save their own configuration.

If this feature is not desired then enter NONE in the window

Use Load Special Area File to load a predefined set of data

Use Save Special Area File to save the currently displayed setting

## Area Definitions

### Area N

Comment: Optional. Enter a comment to describe the area being specfied.

### Corners:

Also called nodes or sides. This is limited by Max Nodes in Area. See SHOW Args for current values. This can be changed on the command line. See above

### Corner N

These are the coordinates of the area vertices. Enter the Longitude and Latitude of the vertices for the area. It is up to the user to ensure that a closed shape is defined.

# Namespace Index

## Package List

Here are the packages with brief descriptions (if available):

**AreaManager**  19

**EditMathSetupFrame**  20

**FishMortBySpecAcc**  21

**GeoSams**  22

**Globals**  23

**GrowthFrame**  27

**MainInputFrame**  28

**PointInPolygon**  29

**shapefile**  30

**ShapeTest**  35

**SortByAreaFrame**  37

**SortByRegionFrame**  38

**SortIntoColumns**  39

**SpecialAreaFrame**  42

**Widgets**  43

# Hierarchical Index

## Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

array.array

shapefile.\_Array 44

SortIntoColumns.Column 54

AreaManager.Corner 55

Exception

shapefile.ShapefileException 109

tk.Frame

AreaManager.AreaMgrSubFrame 51

Widgets.ScrollFrame 103

Widgets.SubFrameElement 131

Widgets.SubFrameInterpFunction 133

Widgets.SubFrameXY 135

ttk.Frame

AreaManager.AreaManager 48

EditMathSetupFrame.EditMathSetup 56

FishMortBySpecAcc.FishMortBySpecAcc 59

GrowthFrame.Growth 66

MainInputFrame.MainInput 84

SortByAreaFrame.SortByArea 114

SortByRegionFrame.SortByRegion 121

SpecialAreaFrame.SpecialArea 127

ShapeTest.GeoShape 63

SortByRegionFrame.GeoShape 64

SortIntoColumns.GeoShape 65

list

shapefile.ShapeRecords 112

shapefile.Shapes 113

shapefile.\_Record 45

object

shapefile.Reader 93

shapefile.Shape 106

shapefile.ShapeRecord 110

shapefile.Writer 137

tk.Tk

GeoSams.MainApplication 75

GrowthFrame.Growth 66

# Class Index

## Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

**shapefile.\_Array**  44

**shapefile.\_Record**  45

**AreaManager.AreaManager (This class is used to paint area grouped by )** 48

**AreaManager.AreaMgrSubFrame**  51

**SortIntoColumns.Column**  54

**AreaManager.Corner (Defines floating point data for corner defintions )** 55

**EditMathSetupFrame.EditMathSetup (This class allows the user to edit the Matlab/Octave setup files to fit their environment )** 56

**FishMortBySpecAcc.FishMortBySpecAcc (This class is used to assist the user in defining areas of interest to assess accumulated parameters located in these areas of interest )** 59

**ShapeTest.GeoShape**  63

**SortByRegionFrame.GeoShape (This class is used to define the shape of the regional data )** 64

**SortIntoColumns.GeoShape**  65

**GrowthFrame.Growth (This class allows the user to adjust parameters used in computing scallop growth )** 66

**GeoSams.MainApplication (This class is the parent class for the GUI )** 75

**MainInputFrame.MainInput (This class displays information about GeoSAMS simulation )** 84

**shapefile.Reader**  93

**Widgets.ScrollFrame (Scrollable Frame Class from https://gist.github.com/mp035/9f2027c3ef9172264532fcd6262f3b01 )** 103

**shapefile.Shape**  106

**shapefile.ShapefileException**  109

**shapefile.ShapeRecord**  110

**shapefile.ShapeRecords**  112

**shapefile.Shapes**  113

**SortByAreaFrame.SortByArea (This class is used to assist the user in defining areas of interest to assess accumulated parameters located in these areas of interest )** 114

**SortByRegionFrame.SortByRegion (This class is used to assist the user in defining areas of interest to assess accumulated parameters located in these areas of interest )** 121

**SpecialAreaFrame.SpecialArea (This class is used to assist the user in defining areas of interest to assess accumulated parameters located in these areas of interest )** 127

**Widgets.SubFrameElement (Generic Element )** 131

**Widgets.SubFrameInterpFunction**  133

**Widgets.SubFrameXY (Widget for XY label and entery )** 135

**shapefile.Writer**  137

# File Index

## File List

Here is a list of all files with brief descriptions:

**AreaManager.py**  146

**EditMathSetupFrame.py**  147

**FishMortBySpecAcc.py**  148

**GeoSams.py**  149

**Globals.py**  150

**GrowthFrame.py**  151

**MainInputFrame.py**  152

**PointInPolygon.py**  153

**ShapeTest.py**  156

**SortByAreaFrame.py**  157

**SortByRegionFrame.py**  158

**SortIntoColumns.py**  159

**SpecialAreaFrame.py**  160

**Widgets.py**  161

**PyshpMaster/shapefile.py**  154

# Namespace Documentation

## AreaManager Namespace Reference

### Classes

class **AreaManager***This class is used to paint area grouped by.*

class **AreaMgrSubFrame**class **Corner**

*Defines floating point data for corner defintions.*

## EditMathSetupFrame Namespace Reference

### Classes

class **EditMathSetup***This class allows the user to edit the Matlab/Octave setup files to fit their environment.*

## FishMortBySpecAcc Namespace Reference

### Classes

class **FishMortBySpecAcc***This class is used to assist the user in defining areas of interest to assess accumulated parameters located in these areas of interest.*

## GeoSams Namespace Reference

### Classes

class **MainApplication***This class is the parent class for the GUI.*

### Functions

**ComputeResiduals** (obsFile, gridFile, procID, retDict)

**main** ()

### Function Documentation

#### GeoSams.ComputeResiduals ( *obsFile*, *gridFile*, *procID*, *retDict*)

#### GeoSams.main ()

#### 

## Globals Namespace Reference

### Functions

**DetermineUnitsScale** (desiredParam)

**UpdateEntry** (entry, val)

**ShowMessage** (heading, message, type='info', timeout=2500)

*This method will display the message and then go away after the default time.*

### Variables

str **analDir** = 'Analysis'

str **configDir** = 'Configuration'

str **dataDir** = 'Data'

str **gridDir** = 'Grids'

str **interCfgDir** = 'Interpolation'

str **resultsDir** = 'Results'

str **shapeFileDir** = 'Shapefiles'

str **simCfgDir** = 'Simulation'

str **specAccCfgDir** = 'SpecialAccess'

str **surveyDataDir** = 'OriginalData'

list **comboTFStr** = ['T', 'F']

list **cornerLabelArr** = ['Corner', 'Long', 'Lat ', '0.0', '0.0']

int **frameWidth** = 400

int **frameHeight** = 200

int **scrollFrameHeight** = 600

int **helpXoffset** = 700

int **helpYoffset** = 50

int **meters\_per\_naut\_mile** = 1852

int **grid\_area\_sqm** = **meters\_per\_naut\_mile**\*\*2

str **ABUN** = 'ABUN\_'

str **BIOM** = 'BIOM\_'

str **EBMS** = 'EBMS\_'

str **FEFF** = 'FEFF\_'

str **FMOR** = 'FMOR\_'

str **LAND** = 'LAND\_'

str **LNDW** = 'LNDW\_'

str **LPUE** = 'LPUE\_'

str **RECR** = 'RECR\_'

int **scrollFrameWidth** = 900

str **geometryStr** = '920x725+10+10'

### Function Documentation

#### Globals.DetermineUnitsScale ( *desiredParam*)

#### Globals.ShowMessage ( *heading*, *message*, *type* = 'info', *timeout* = 2500)

This method will display the message and then go away after the default time.

#### Globals.UpdateEntry ( *entry*, *val*)

### Variable Documentation

#### str Globals.ABUN = 'ABUN\_'

#### str Globals.analDir = 'Analysis'

#### str Globals.BIOM = 'BIOM\_'

#### list Globals.comboTFStr = ['T', 'F']

#### str Globals.configDir = 'Configuration'

#### list Globals.cornerLabelArr = ['Corner', 'Long', 'Lat ', '0.0', '0.0']

#### str Globals.dataDir = 'Data'

#### str Globals.EBMS = 'EBMS\_'

#### str Globals.FEFF = 'FEFF\_'

#### str Globals.FMOR = 'FMOR\_'

#### int Globals.frameHeight = 200

#### int Globals.frameWidth = 400

#### str Globals.geometryStr = '920x725+10+10'

#### int Globals.grid\_area\_sqm = meters\_per\_naut\_mile\*\*2

#### str Globals.gridDir = 'Grids'

#### int Globals.helpXoffset = 700

#### int Globals.helpYoffset = 50

#### str Globals.interCfgDir = 'Interpolation'

#### str Globals.LAND = 'LAND\_'

#### str Globals.LNDW = 'LNDW\_'

#### str Globals.LPUE = 'LPUE\_'

#### int Globals.meters\_per\_naut\_mile = 1852

#### str Globals.RECR = 'RECR\_'

#### str Globals.resultsDir = 'Results'

#### int Globals.scrollFrameHeight = 600

#### int Globals.scrollFrameWidth = 900

#### str Globals.shapeFileDir = 'Shapefiles'

#### str Globals.simCfgDir = 'Simulation'

#### str Globals.specAccCfgDir = 'SpecialAccess'

#### str Globals.surveyDataDir = 'OriginalData'

#### 

## GrowthFrame Namespace Reference

### Classes

class **Growth***This class allows the user to adjust parameters used in computing scallop growth.*

## MainInputFrame Namespace Reference

### Classes

class **MainInput***This class displays information about GeoSAMS simulation.*

## PointInPolygon Namespace Reference

### Functions

**PointInPolygon** (polyX, polyY, x, y, nodes)

### Function Documentation

#### PointInPolygon.PointInPolygon ( *polyX*, *polyY*, *x*, *y*, *nodes*)

@param float polyX Array of horizontal, coordinates of corners

@param float polyY Array of vertical coordinates of corners

@param float x horizontal coordinate of point we wish to determine if inside polygram

@param float y vertical coordinate of point we wish to determine if inside polygram

@param int nodes the number of corners, edges, that define the polygon

@returns true if point is inside polygram or if on vert or horiz edge,

false if outside

if point is on rise of falling edge then it may return true or false

## shapefile Namespace Reference

### Classes

class **\_Array**class **\_Record**

class **Reader**

class **Shape**

class **ShapefileException**

class **ShapeRecord**

class **ShapeRecords**

class **Shapes**

class **Writer**

### Functions

**b** (v, encoding='utf-8', encodingErrors='strict')

**u** (v, encoding='utf-8', encodingErrors='strict')

**is\_string** (v)

**pathlike\_obj** (path)

**signed\_area** (coords, fast=False)

**is\_cw** (coords)

**rewind** (coords)

**ring\_bbox** (coords)

**bbox\_overlap** (bbox1, bbox2)

**bbox\_contains** (bbox1, bbox2)

**ring\_contains\_point** (coords, p)

**ring\_sample** (coords, ccw=False)

**ring\_contains\_ring** (coords1, coords2)

**organize\_polygon\_rings** (rings, return\_errors=None)

**test** (\*\*kwargs)

### Variables

str **\_\_version\_\_** = "2.3.1"

**logger** = logging.getLogger(\_\_name\_\_)

bool **VERBOSE** = True

int **NULL** = 0

int **POINT** = 1

int **POLYLINE** = 3

int **POLYGON** = 5

int **MULTIPOINT** = 8

int **POINTZ** = 11

int **POLYLINEZ** = 13

int **POLYGONZ** = 15

int **MULTIPOINTZ** = 18

int **POINTM** = 21

int **POLYLINEM** = 23

int **POLYGONM** = 25

int **MULTIPOINTM** = 28

int **MULTIPATCH** = 31

dict **SHAPETYPE\_LOOKUP**

int **TRIANGLE\_STRIP** = 0

int **TRIANGLE\_FAN** = 1

int **OUTER\_RING** = 2

int **INNER\_RING** = 3

int **FIRST\_RING** = 4

int **RING** = 5

dict **PARTTYPE\_LOOKUP**

int **PYTHON3** = 3

**xrange** = range

**izip** = zip

list **MISSING** = [None,'']

int **NODATA** = -10e38

**failure\_count** = **test**()

### Detailed Description

shapefile.py

Provides read and write support for ESRI Shapefiles.

authors: jlawhead<at>geospatialpython.com

maintainer: karim.bahgat.norway<at>gmail.com

Compatible with Python versions 2.7-3.x

### Function Documentation

#### shapefile.b ( *v*, *encoding* = 'utf-8', *encodingErrors* = 'strict')

#### shapefile.bbox\_contains ( *bbox1*, *bbox2*)

Tests whether bbox1 fully contains bbox2, returning a boolean

#### shapefile.bbox\_overlap ( *bbox1*, *bbox2*)

Tests whether two bounding boxes overlap, returning a boolean

#### shapefile.is\_cw ( *coords*)

Returns True if a polygon ring has clockwise orientation, determined

by a negatively signed area.

#### shapefile.is\_string ( *v*)

#### shapefile.organize\_polygon\_rings ( *rings*, *return\_errors* = None)

Organize a list of coordinate rings into one or more polygons with holes.

Returns a list of polygons, where each polygon is composed of a single exterior

ring, and one or more interior holes. If a return\_errors dict is provided (optional),

any errors encountered will be added to it.

Rings must be closed, and cannot intersect each other (non-self-intersecting polygon).

Rings are determined as exteriors if they run in clockwise direction, or interior

holes if they run in counter-clockwise direction. This method is used to construct

GeoJSON (multi)polygons from the shapefile polygon shape type, which does not

explicitly store the structure of the polygons beyond exterior/interior ring orientation.

#### shapefile.pathlike\_obj ( *path*)

#### shapefile.rewind ( *coords*)

Returns the input coords in reversed order.

#### shapefile.ring\_bbox ( *coords*)

Calculates and returns the bounding box of a ring.

#### shapefile.ring\_contains\_point ( *coords*, *p*)

Fast point-in-polygon crossings algorithm, MacMartin optimization.

Adapted from code by Eric Haynes

http://www.realtimerendering.com/resources/GraphicsGems//gemsiv/ptpoly\_haines/ptinpoly.c

Original description:

Shoot a test ray along +X axis. The strategy, from MacMartin, is to

compare vertex Y values to the testing point's Y and quickly discard

edges which are entirely to one side of the test ray.

#### shapefile.ring\_contains\_ring ( *coords1*, *coords2*)

Returns True if all vertexes in coords2 are fully inside coords1.

#### shapefile.ring\_sample ( *coords*, *ccw* = False)

Return a sample point guaranteed to be within a ring, by efficiently

finding the first centroid of a coordinate triplet whose orientation

matches the orientation of the ring and passes the point-in-ring test.

The orientation of the ring is assumed to be clockwise, unless ccw

(counter-clockwise) is set to True.

#### shapefile.signed\_area ( *coords*, *fast* = False)

Return the signed area enclosed by a ring using the linear time

algorithm. A value >= 0 indicates a counter-clockwise oriented ring.

A faster version is possible by setting 'fast' to True, which returns

2x the area, e.g. if you're only interested in the sign of the area.

#### shapefile.test (\*\* *kwargs*)

#### shapefile.u ( *v*, *encoding* = 'utf-8', *encodingErrors* = 'strict')

### Variable Documentation

#### str shapefile.\_\_version\_\_ = "2.3.1"[private]

#### shapefile.failure\_count = test()

#### int shapefile.FIRST\_RING = 4

#### int shapefile.INNER\_RING = 3

#### shapefile.izip = zip

#### shapefile.logger = logging.getLogger(\_\_name\_\_)

#### list shapefile.MISSING = [None,'']

#### int shapefile.MULTIPATCH = 31

#### int shapefile.MULTIPOINT = 8

#### int shapefile.MULTIPOINTM = 28

#### int shapefile.MULTIPOINTZ = 18

#### int shapefile.NODATA = -10e38

#### int shapefile.NULL = 0

#### int shapefile.OUTER\_RING = 2

#### dict shapefile.PARTTYPE\_LOOKUP

**Initial value:**1 = {

2 0: 'TRIANGLE\_STRIP',

3 1: 'TRIANGLE\_FAN',

4 2: 'OUTER\_RING',

5 3: 'INNER\_RING',

6 4: 'FIRST\_RING',

7 5: 'RING'}

#### int shapefile.POINT = 1

#### int shapefile.POINTM = 21

#### int shapefile.POINTZ = 11

#### int shapefile.POLYGON = 5

#### int shapefile.POLYGONM = 25

#### int shapefile.POLYGONZ = 15

#### int shapefile.POLYLINE = 3

#### int shapefile.POLYLINEM = 23

#### int shapefile.POLYLINEZ = 13

#### int shapefile.PYTHON3 = 3

#### int shapefile.RING = 5

#### dict shapefile.SHAPETYPE\_LOOKUP

**Initial value:**1 = {

2 0: 'NULL',

3 1: 'POINT',

4 3: 'POLYLINE',

5 5: 'POLYGON',

6 8: 'MULTIPOINT',

7 11: 'POINTZ',

8 13: 'POLYLINEZ',

9 15: 'POLYGONZ',

10 18: 'MULTIPOINTZ',

11 21: 'POINTM',

12 23: 'POLYLINEM',

13 25: 'POLYGONM',

14 28: 'MULTIPOINTM',

15 31: 'MULTIPATCH'}

#### int shapefile.TRIANGLE\_FAN = 1

#### int shapefile.TRIANGLE\_STRIP = 0

#### bool shapefile.VERBOSE = True

#### shapefile.xrange = range

#### 

## ShapeTest Namespace Reference

### Classes

### class GeoShapeVariables

**sf** = **shapefile.Reader**("Shapefiles/MAB\_Estimation\_Areas\_2019\_UTM18\_PDT.shp")

**shapes** = sf.shapes()

**shapeLen** = len(**sf**)

list **shapeMA** = [ **GeoShape**() for \_ in range(**shapeLen**)]

**record** = sf.record(n)

**as\_dict** = record.as\_dict()

**SAMS**

**NewSAMS**

**areaKm2**

**pointLen** = len(**shapes**[n].points)

**X**

**Y**

**lat**

**lon**

list **shapeGB** = [ **GeoShape**() for \_ in range(**shapeLen**)]

### Variable Documentation

#### ShapeTest.areaKm2

#### ShapeTest.as\_dict = record.as\_dict()

#### ShapeTest.lat

#### ShapeTest.lon

#### ShapeTest.NewSAMS

#### ShapeTest.pointLen = len(shapes[n].points)

#### ShapeTest.record = sf.record(n)

#### ShapeTest.SAMS

#### ShapeTest.sf = shapefile.Reader("Shapefiles/MAB\_Estimation\_Areas\_2019\_UTM18\_PDT.shp")

#### list ShapeTest.shapeGB = [ GeoShape() for \_ in range(shapeLen)]

#### ShapeTest.shapeLen = len(sf)

#### list ShapeTest.shapeMA = [ GeoShape() for \_ in range(shapeLen)]

#### ShapeTest.shapes = sf.shapes()

#### ShapeTest.X

#### ShapeTest.Y

#### 

## SortByAreaFrame Namespace Reference

### Classes

class **SortByArea***This class is used to assist the user in defining areas of interest to assess accumulated parameters located in these areas of interest.*

## SortByRegionFrame Namespace Reference

### Classes

class **GeoShape***This class is used to define the shape of the regional data.*

class **SortByRegion***This class is used to assist the user in defining areas of interest to assess accumulated parameters located in these areas of interest.*

## SortIntoColumns Namespace Reference

### Classes

class **Column**class **GeoShape**

### Variables

**inputFile** = sys.argv[1]

**l** = len(**inputFile**)

**domain** = **inputFile**[**l**-2:**l**]

**dataFile** = os.path.join('Data', **inputFile**+'.csv')

**outfile** = os.path.join('Data', **inputFile**+'\_BUFFER.csv')

**M** = pd.read\_csv(**dataFile**)

**fileName** = os.environ['GBShapeBufferFile']

str **subDir** = 'GB\_Buffer'

**shapeFile** = os.path.join('Shapefiles', **subDir**, **fileName**)

**sf** = **shapefile.Reader**(**shapeFile**)

**shapes** = sf.shapes()

**shapeLen** = len(**sf**)

list **shape** = [ **GeoShape**() for \_ in range(**shapeLen**)]

**record** = sf.record(n)

**as\_dict** = record.as\_dict()

**Region**

**pointLen** = len(**shapes**[n].points)

**X**

**Y**

list **columns** = [**Column**() for \_ in range(**shapeLen**)]

**name**

**X\_t** = **M**['UTM\_X']

**Y\_t** = **M**['UTM\_Y']

**rows** = len(**X\_t**)

**nodes** = len(**shape**[rgn].**X**)

**sep**

**na\_rep**

**index**

### Variable Documentation

#### SortIntoColumns.as\_dict = record.as\_dict()

#### list SortIntoColumns.columns = [Column() for \_ in range(shapeLen)]

#### SortIntoColumns.dataFile = os.path.join('Data', inputFile+'.csv')

#### SortIntoColumns.domain = inputFile[l-2:l]

#### SortIntoColumns.fileName = os.environ['GBShapeBufferFile']

#### SortIntoColumns.index

#### SortIntoColumns.inputFile = sys.argv[1]

#### SortIntoColumns.l = len(inputFile)

#### SortIntoColumns.M = pd.read\_csv(dataFile)

#### SortIntoColumns.na\_rep

#### SortIntoColumns.name

#### SortIntoColumns.nodes = len(shape[rgn].X)

#### SortIntoColumns.outfile = os.path.join('Data', inputFile+'\_BUFFER.csv')

#### SortIntoColumns.pointLen = len(shapes[n].points)

#### SortIntoColumns.record = sf.record(n)

#### SortIntoColumns.Region

#### SortIntoColumns.rows = len(X\_t)

#### SortIntoColumns.sep

#### SortIntoColumns.sf = shapefile.Reader(shapeFile)

#### list SortIntoColumns.shape = [ GeoShape() for \_ in range(shapeLen)]

#### SortIntoColumns.shapeFile = os.path.join('Shapefiles', subDir, fileName)

#### SortIntoColumns.shapeLen = len(sf)

#### SortIntoColumns.shapes = sf.shapes()

#### str SortIntoColumns.subDir = 'GB\_Buffer'

#### SortIntoColumns.X

#### SortIntoColumns.X\_t = M['UTM\_X']

#### SortIntoColumns.Y

#### SortIntoColumns.Y\_t = M['UTM\_Y']

#### 

## SpecialAreaFrame Namespace Reference

### Classes

class **SpecialArea***This class is used to assist the user in defining areas of interest to assess accumulated parameters located in these areas of interest.*

## Widgets Namespace Reference

### Classes

class **ScrollFrame***Scrollable Frame Class from https://gist.github.com/mp035/9f2027c3ef9172264532fcd6262f3b01.*

class **SubFrameElement***Generic Element.*

class **SubFrameInterpFunction**class **SubFrameXY**

*Widget for XY label and entery.*

### Functions

**numbersCallback** (input)

*Allows only correctly formed positive integers, ignores non-numeric characters.*

**floatCallback** (input)

*Allows only correctly formed floats, ignores non-numeric characters.*

### Function Documentation

#### Widgets.floatCallback ( *input*)

Allows only correctly formed floats, ignores non-numeric characters.

#### Widgets.numbersCallback ( *input*)

Allows only correctly formed positive integers, ignores non-numeric characters.

Only allows numeric for input

# Class Documentation

## shapefile.\_Array Class Reference

Inheritance diagram for shapefile.\_Array:

### Public Member Functions

**\_\_repr\_\_** (self)

### Detailed Description

Converts python tuples to lists of the appropriate type.

Used to unpack different shapefile header parts.

### Member Function Documentation

#### shapefile.\_Array.\_\_repr\_\_ ( *self*)

#### The documentation for this class was generated from the following file:

PyshpMaster/**shapefile.py**

## shapefile.\_Record Class Reference

Inheritance diagram for shapefile.\_Record:

### Public Member Functions

**\_\_init\_\_** (self, field\_positions, values, **oid**=None)

**\_\_getattr\_\_** (self, item)

**\_\_setattr\_\_** (self, key, value)

**\_\_getitem\_\_** (self, item)

**\_\_setitem\_\_** (self, key, value)

**oid** (self)

**as\_dict** (self, date\_strings=False)

**\_\_repr\_\_** (self)

**\_\_dir\_\_** (self)

### Private Attributes

**\_\_field\_positions\_\_oid**

### Detailed Description

A class to hold a record. Subclasses list to ensure compatibility with

former work and to reuse all the optimizations of the builtin list.

In addition to the list interface, the values of the record

can also be retrieved using the field's name. For example if the dbf contains

a field ID at position 0, the ID can be retrieved with the position, the field name

as a key, or the field name as an attribute.

>>> # Create a Record with one field, normally the record is created by the Reader class

>>> r = \_Record({'ID': 0}, [0])

>>> print(r[0])

>>> print(r['ID'])

>>> print(r.ID)

### Constructor & Destructor Documentation

#### shapefile.\_Record.\_\_init\_\_ ( *self*, *field\_positions*, *values*, *oid* = None)

A Record should be created by the Reader class

:param field\_positions: A dict mapping field names to field positions

:param values: A sequence of values

:param oid: The object id, an int (optional)

### Member Function Documentation

#### shapefile.\_Record.\_\_dir\_\_ ( *self*)

Helps to show the field names in an interactive environment like IPython.

See: http://ipython.readthedocs.io/en/stable/config/integrating.html

:return: List of method names and fields

#### shapefile.\_Record.\_\_getattr\_\_ ( *self*, *item*)

\_\_getattr\_\_ is called if an attribute is used that does

not exist in the normal sense. For example r=Record(...), r.ID

calls r.\_\_getattr\_\_('ID'), but r.index(5) calls list.index(r, 5)

:param item: The field name, used as attribute

:return: Value of the field

:raises: AttributeError, if item is not a field of the shapefile

and IndexError, if the field exists but the field's

corresponding value in the Record does not exist

#### shapefile.\_Record.\_\_getitem\_\_ ( *self*, *item*)

Extends the normal list item access with

access using a fieldname

For example r['ID'], r[0]

:param item: Either the position of the value or the name of a field

:return: the value of the field

#### shapefile.\_Record.\_\_repr\_\_ ( *self*)

#### shapefile.\_Record.\_\_setattr\_\_ ( *self*, *key*, *value*)

Sets a value of a field attribute

:param key: The field name

:param value: the value of that field

:return: None

:raises: AttributeError, if key is not a field of the shapefile

#### shapefile.\_Record.\_\_setitem\_\_ ( *self*, *key*, *value*)

Extends the normal list item access with

access using a fieldname

For example r['ID']=2, r[0]=2

:param key: Either the position of the value or the name of a field

:param value: the new value of the field

#### shapefile.\_Record.as\_dict ( *self*, *date\_strings* = False)

Returns this Record as a dictionary using the field names as keys

:return: dict

#### shapefile.\_Record.oid ( *self*)

The index position of the record in the original shapefile

### Member Data Documentation

#### shapefile.\_Record.\_\_field\_positions[private]

#### shapefile.\_Record.\_\_oid[private]

#### The documentation for this class was generated from the following file:

PyshpMaster/**shapefile.py**

## AreaManager.AreaManager Class Reference

This class is used to paint area grouped by.

Inheritance diagram for AreaManager.AreaManager:

### Public Member Functions

**\_\_init\_\_** (self, container, parent, maxAreas, maxCorners, elementRow, elementCol, **cornerRow**, cornerColumn, labelArr, includeYears=False, numYearsMax=0, yearStart=0, yearStop=0, includeArea=False)

**NumAreasUpdate** (self, **numAreas**)

**ReadAreaCorners** (self, fName)

**UpdateWidgets** (self, showCompArea=True)

### Public Attributes

**numAreasMaxnumCornersMax**

**numAreas**

**numCorners**

**cornerRow**

**areaData**

**areaSubFrame**

**scrollFrame**

**sortAreaFrame**

### Detailed Description

This class is used to paint area grouped by.

Area N Comment Number of Nodes Update Nodes Node 1 .... Node N X data .... X data Y data .... Y data

### Constructor & Destructor Documentation

#### AreaManager.AreaManager.\_\_init\_\_ ( *self*, *container*, *parent*, *maxAreas*, *maxCorners*, *elementRow*, *elementCol*, *cornerRow*, *cornerColumn*, *labelArr*, *includeYears* = False, *numYearsMax* = 0, *yearStart* = 0, *yearStop* = 0, *includeArea* = False)

### Member Function Documentation

#### AreaManager.AreaManager.NumAreasUpdate ( *self*, *numAreas*)

Updates the number of areas functions.

#### AreaManager.AreaManager.ReadAreaCorners ( *self*, *fName*)

Reads an Area file and returns the number of nodes defined

#### AreaManager.AreaManager.UpdateWidgets ( *self*, *showCompArea* = True)

### Member Data Documentation

#### AreaManager.AreaManager.areaData

#### AreaManager.AreaManager.areaSubFrame

#### AreaManager.AreaManager.cornerRow

#### AreaManager.AreaManager.numAreas

#### AreaManager.AreaManager.numAreasMax

#### AreaManager.AreaManager.numCorners

#### AreaManager.AreaManager.numCornersMax

#### AreaManager.AreaManager.scrollFrame

#### AreaManager.AreaManager.sortAreaFrame

#### The documentation for this class was generated from the following file:

#### AreaManager.py

## AreaManager.AreaMgrSubFrame Class Reference

Inheritance diagram for AreaManager.AreaMgrSubFrame:

### Public Member Functions

**\_\_init\_\_** (self, container, parent, areaNum, **numCorners**, **numCornersMax**, elementRow, elementCol, cornerRow, cornerCol, labelArr, includeYears=False, **numYearsMax**=0, **yearStart**=0, **yearStop**=0, includeArea=False)

**SaveField** (self, event)

**EnterKeyClicked** (self, event)

**NumCornersUpdate** (self)

**AppendResults** (self, addYears)

*This method is used to add results when the original maximum number of years is exceeded.*

### Public Attributes

**numCornersMaxstartincCol**

**yearStart**

**yearStop**

**numYearsMax**

**comment**

**myArea**

**areaFrame**

**numYrCols**

**results**

**commentEntry**

**compAreaEntry**

**numCornersEntry**

**corners**

**numCorners**

### Constructor & Destructor Documentation

#### AreaManager.AreaMgrSubFrame.\_\_init\_\_ ( *self*, *container*, *parent*, *areaNum*, *numCorners*, *numCornersMax*, *elementRow*, *elementCol*, *cornerRow*, *cornerCol*, *labelArr*, *includeYears* = False, *numYearsMax* = 0, *yearStart* = 0, *yearStop* = 0, *includeArea* = False)

### Member Function Documentation

#### AreaManager.AreaMgrSubFrame.AppendResults ( *self*, *addYears*)

This method is used to add results when the original maximum number of years is exceeded.

#### AreaManager.AreaMgrSubFrame.EnterKeyClicked ( *self*, *event*)

#### AreaManager.AreaMgrSubFrame.NumCornersUpdate ( *self*)

#### AreaManager.AreaMgrSubFrame.SaveField ( *self*, *event*)

### Member Data Documentation

#### AreaManager.AreaMgrSubFrame.areaFrame

#### AreaManager.AreaMgrSubFrame.comment

#### AreaManager.AreaMgrSubFrame.commentEntry

#### AreaManager.AreaMgrSubFrame.compAreaEntry

#### AreaManager.AreaMgrSubFrame.corners

#### AreaManager.AreaMgrSubFrame.myArea

#### AreaManager.AreaMgrSubFrame.numCorners

#### AreaManager.AreaMgrSubFrame.numCornersEntry

#### AreaManager.AreaMgrSubFrame.numCornersMax

#### AreaManager.AreaMgrSubFrame.numYearsMax

#### AreaManager.AreaMgrSubFrame.numYrCols

#### AreaManager.AreaMgrSubFrame.results

#### AreaManager.AreaMgrSubFrame.startincCol

#### AreaManager.AreaMgrSubFrame.yearStart

#### AreaManager.AreaMgrSubFrame.yearStop

#### The documentation for this class was generated from the following file:

#### AreaManager.py

## SortIntoColumns.Column Class Reference

### Public Member Functions

**\_\_init\_\_** (self)

### Public Attributes

**nameinBox**

### Constructor & Destructor Documentation

#### SortIntoColumns.Column.\_\_init\_\_ ( *self*)

### Member Data Documentation

#### SortIntoColumns.Column.inBox

#### SortIntoColumns.Column.name

#### The documentation for this class was generated from the following file:

#### SortIntoColumns.py

## AreaManager.Corner Class Reference

Defines floating point data for corner defintions.

### Public Member Functions

**\_\_init\_\_** (self, maxCorners)

### Public Attributes

**longlat**

**numCorners**

### Detailed Description

Defines floating point data for corner defintions.

long, lat have become interchangeable with x, y

### Constructor & Destructor Documentation

#### AreaManager.Corner.\_\_init\_\_ ( *self*, *maxCorners*)

### Member Data Documentation

#### AreaManager.Corner.lat

#### AreaManager.Corner.long

#### AreaManager.Corner.numCorners

#### The documentation for this class was generated from the following file:

#### AreaManager.py

## EditMathSetupFrame.EditMathSetup Class Reference

This class allows the user to edit the Matlab/Octave setup files to fit their environment.

Inheritance diagram for EditMathSetupFrame.EditMathSetup:

### Public Member Functions

**\_\_init\_\_** (self, container)

*Constructor for Growth Class.*

**on\_visibility** (self, event)

*Opens either startup.m or .octaverc depending if user selected Matlab or Octave resp.*

**LoadStartupFile** (self)

**ReadFile** (self, showMsg)

**WriteStartupFile** (self)

**pop\_up** (self)

### Public Attributes

**matlabFNameoctaveFName**

**editMathFrame**

**editText**

**editTextButton**

**isMatlab**

**usingMatlab**

**useMatlabRB**

**useOctaveRB**

**on\_visibility**

### Detailed Description

This class allows the user to edit the Matlab/Octave setup files to fit their environment.

### Constructor & Destructor Documentation

#### EditMathSetupFrame.EditMathSetup.\_\_init\_\_ ( *self*, *container*)

Constructor for Growth Class.

### Member Function Documentation

#### EditMathSetupFrame.EditMathSetup.LoadStartupFile ( *self*)

#### EditMathSetupFrame.EditMathSetup.on\_visibility ( *self*, *event*)

Opens either startup.m or .octaverc depending if user selected Matlab or Octave resp.

#### EditMathSetupFrame.EditMathSetup.pop\_up ( *self*)

#### EditMathSetupFrame.EditMathSetup.ReadFile ( *self*, *showMsg*)

#### EditMathSetupFrame.EditMathSetup.WriteStartupFile ( *self*)

### Member Data Documentation

#### EditMathSetupFrame.EditMathSetup.editMathFrame

#### EditMathSetupFrame.EditMathSetup.editText

#### EditMathSetupFrame.EditMathSetup.editTextButton

#### EditMathSetupFrame.EditMathSetup.isMatlab

#### EditMathSetupFrame.EditMathSetup.matlabFName

#### EditMathSetupFrame.EditMathSetup.octaveFName

#### EditMathSetupFrame.EditMathSetup.on\_visibility

#### EditMathSetupFrame.EditMathSetup.useMatlabRB

#### EditMathSetupFrame.EditMathSetup.useOctaveRB

#### EditMathSetupFrame.EditMathSetup.usingMatlab

#### The documentation for this class was generated from the following file:

#### EditMathSetupFrame.py

## FishMortBySpecAcc.FishMortBySpecAcc Class Reference

This class is used to assist the user in defining areas of interest to assess accumulated parameters located in these areas of interest.

Inheritance diagram for FishMortBySpecAcc.FishMortBySpecAcc:

### Public Member Functions

**\_\_init\_\_** (self, container, maxAreas, maxCorners)

**on\_visibility** (self, event)

**EnterKeyClicked** (self, event)

**NumDefinedUpdate** (self)

**UpdateWidgets** (self)

**GetFMFile** (self)

**SaveFMFile** (self)

**ReadFields** (self, fName)

**pop\_up** (self)

*Help Window for Fishing Mortatlity in Special Access Area.*

### Public Attributes

**rootstartDir**

**fmFName**

**numDefinedMax**

**numFieldsMax**

**numDefined**

**numFieldss**

**numDefinedLabel**

**numDefinedEntry**

**EnterKeyClicked**

**fishMortFile**

**fishMortFileLabel**

**openFMFileButton**

**saveFMFileButton**

**yearEntry**

**areaMgr**

**on\_visibility**

**numAreas**

### Detailed Description

This class is used to assist the user in defining areas of interest to assess accumulated parameters located in these areas of interest.

### Constructor & Destructor Documentation

#### FishMortBySpecAcc.FishMortBySpecAcc.\_\_init\_\_ ( *self*, *container*, *maxAreas*, *maxCorners*)

### Member Function Documentation

#### FishMortBySpecAcc.FishMortBySpecAcc.EnterKeyClicked ( *self*, *event*)

#### FishMortBySpecAcc.FishMortBySpecAcc.GetFMFile ( *self*)

#### FishMortBySpecAcc.FishMortBySpecAcc.NumDefinedUpdate ( *self*)

#### FishMortBySpecAcc.FishMortBySpecAcc.on\_visibility ( *self*, *event*)

#### FishMortBySpecAcc.FishMortBySpecAcc.pop\_up ( *self*)

Help Window for Fishing Mortatlity in Special Access Area.

#### FishMortBySpecAcc.FishMortBySpecAcc.ReadFields ( *self*, *fName*)

Reads an Area file and returns the number of fields.

Fields have a Special Area number for the x value with a Mortality setting for the y value.

#### FishMortBySpecAcc.FishMortBySpecAcc.SaveFMFile ( *self*)

#### FishMortBySpecAcc.FishMortBySpecAcc.UpdateWidgets ( *self*)

### Member Data Documentation

#### FishMortBySpecAcc.FishMortBySpecAcc.areaMgr

#### FishMortBySpecAcc.FishMortBySpecAcc.EnterKeyClicked

#### FishMortBySpecAcc.FishMortBySpecAcc.fishMortFile

#### FishMortBySpecAcc.FishMortBySpecAcc.fishMortFileLabel

#### FishMortBySpecAcc.FishMortBySpecAcc.fmFName

#### FishMortBySpecAcc.FishMortBySpecAcc.numAreas

#### FishMortBySpecAcc.FishMortBySpecAcc.numDefined

#### FishMortBySpecAcc.FishMortBySpecAcc.numDefinedEntry

#### FishMortBySpecAcc.FishMortBySpecAcc.numDefinedLabel

#### FishMortBySpecAcc.FishMortBySpecAcc.numDefinedMax

#### FishMortBySpecAcc.FishMortBySpecAcc.numFieldsMax

#### FishMortBySpecAcc.FishMortBySpecAcc.numFieldss

#### FishMortBySpecAcc.FishMortBySpecAcc.on\_visibility

#### FishMortBySpecAcc.FishMortBySpecAcc.openFMFileButton

#### FishMortBySpecAcc.FishMortBySpecAcc.root

#### FishMortBySpecAcc.FishMortBySpecAcc.saveFMFileButton

#### FishMortBySpecAcc.FishMortBySpecAcc.startDir

#### FishMortBySpecAcc.FishMortBySpecAcc.yearEntry

#### The documentation for this class was generated from the following file:

#### FishMortBySpecAcc.py

## ShapeTest.GeoShape Class Reference

### Public Member Functions

**\_\_init\_\_** (self)

### Public Attributes

**XY**

**lat**

**lon**

**SAMS**

**NewSAMS**

**areaKm2**

### Constructor & Destructor Documentation

#### ShapeTest.GeoShape.\_\_init\_\_ ( *self*)

### Member Data Documentation

#### ShapeTest.GeoShape.areaKm2

#### ShapeTest.GeoShape.lat

#### ShapeTest.GeoShape.lon

#### ShapeTest.GeoShape.NewSAMS

#### ShapeTest.GeoShape.SAMS

#### ShapeTest.GeoShape.X

#### ShapeTest.GeoShape.Y

#### The documentation for this class was generated from the following file:

#### ShapeTest.py

## SortByRegionFrame.GeoShape Class Reference

This class is used to define the shape of the regional data.

### Public Member Functions

**\_\_init\_\_** (self)

### Public Attributes

**XY**

**lat**

**lon**

**Zone**

**area**

### Detailed Description

This class is used to define the shape of the regional data.

This class uses PyShp libraries as defined in https://github.com/GeospatialPython/pyshp ==> Code ==> Download ZIP

### Constructor & Destructor Documentation

#### SortByRegionFrame.GeoShape.\_\_init\_\_ ( *self*)

### Member Data Documentation

#### SortByRegionFrame.GeoShape.area

#### SortByRegionFrame.GeoShape.lat

#### SortByRegionFrame.GeoShape.lon

#### SortByRegionFrame.GeoShape.X

#### SortByRegionFrame.GeoShape.Y

#### SortByRegionFrame.GeoShape.Zone

#### The documentation for this class was generated from the following file:

#### SortByRegionFrame.py

## SortIntoColumns.GeoShape Class Reference

### Public Member Functions

**\_\_init\_\_** (self)

### Public Attributes

**XY**

**lat**

**lon**

**Region**

### Constructor & Destructor Documentation

#### SortIntoColumns.GeoShape.\_\_init\_\_ ( *self*)

### Member Data Documentation

#### SortIntoColumns.GeoShape.lat

#### SortIntoColumns.GeoShape.lon

#### SortIntoColumns.GeoShape.Region

#### SortIntoColumns.GeoShape.X

#### SortIntoColumns.GeoShape.Y

#### The documentation for this class was generated from the following file:

#### SortIntoColumns.py

## GrowthFrame.Growth Class Reference

This class allows the user to adjust parameters used in computing scallop growth.

Inheritance diagram for GrowthFrame.Growth:

### Public Member Functions

**\_\_init\_\_** (self, container, **friend**)

*Constructor for Growth Class.*

**on\_visibility** (self, event)

**GetGrowthConfigFName** (self)

*Calls the filedialog method asksaveasfilename to name a file to be used for the Mortality Configuration file.*

**LoadGrowthData** (self)

**UpdateWidgets** (self)

**UpdateValues** (self)

*Method to read Mortality Configuration file and set values accordingly.*

**pop\_up** (self)

#### Public Member Functions inherited from GeoSams.MainApplication

**ToggleSkipStatusMsgs** (self)

**ShowArgs** (self)

*Display setup limits here Messagebox blocks entry widgets if attempted to open before the main window completes.*

**Run\_Sim** (self)

*Starts the GeoSAMS simulatation* ***ScallopPopDensity*** *.*

**InterpAndPlotResults** (self)

**SaveConfigFiles** (self)

*Save all of the defined configuration files.*

**WriteScallopConfig** (self)

*Saves simulation configuration file.*

**ConvertMonthDayToDayOfYr** (self, monthDayStr)

*This method is used to converty the recruitment start and stop dates from a string month numerical day into days in a year.*

**ConvertDayOfYrToMonthDay** (self, dayOfYear)

**ReadRecruitConfigFile** (self)

*Read in the (tag, value) parameters from the recruitment to update parameters.*

**WriteRecruitmentConfig** (self)

*Saves recruitment parameters to a configuration file.*

**WriteGrowthConfig** (self)

*Saves mortality parameters to a configuration file.*

**WriteGridMgrConfig** (self)

*Saves grid manager parameters to a configuration file.*

**WriteSpatialFncsConfig** (self, cfgFile, functions, numFncsEntry)

*Saves spatial function parameters to a configuration file.*

**ReadConfigFile** (self, fName)

*Reads a typical configuration file to recover the tags and values.*

**ReadSimConfigFile** (self)

*Read in the (tag, value) parameters from the simulation configuration file.*

**ReadGridMgrConfigFile** (self)

*Read in the (tag, value) parameters from the grid manager configuration file.*

### Public Attributes

**rootgrowthStartDir**

**friend**

**fmortStr**

**alphStr**

**maCullStr**

**maDiscStr**

**gbCullStr**

**gbDiscStr**

**maFSelAStr**

**maFSelBStr**

**gbClFSelAStr**

**gbClFSelBStr**

**gbOpFSelAStr**

**gbOpFSelBStr**

**maAdultMortStr**

**gbAdultMortStr**

**maIncidStr**

**gbIncidStr**

**maLen0Str**

**gbLen0Str**

**fmortFileName**

**lpueSlStr**

**lpueSl2Str**

**lpueIntcStr**

**maxPerDayStr**

**maxTimeStr**

**dredgeWdStr**

**towSpdStr**

**fishMort**

**alphaMort**

**maAdultMort**

**gbAdultMort**

**maLength0**

**gbLength0**

**maFSelectA**

**maFSelectB**

**gbClosedFSelectA**

**gbClosedFSelectB**

**gbOpenFSelectA**

**gbOpenFSelectB**

**lpueSlope**

**lpueSlope2**

**lpueIntcept**

**maxPerDay**

**maxTime**

**dredgeWth**

**towSpeed**

**maIncident**

**gbIncident**

**maCullSize**

**maDiscard**

**gbCullSize**

**gbDiscard**

**growthCfgFile**

**saveGrowthConfigButton**

**loadGrowthConfigButton**

**on\_visibility**

#### Public Attributes inherited from GeoSams.MainApplication

**monDict**

**daysInYear**

**maxAreas**

**maxCorners**

**maxYears**

**addFrameClicked**

**tsPerYear**

**paramVal**

**paramStr**

**useHabCamData**

**domainName**

**yearStart**

**yearStop**

**simConfigFile**

**style**

**root**

**notebook**

**frame1**

**recrConfigFile**

**frame2**

**frame3**

**frame5**

**frame6**

**frame7**

**frame8**

**gmConfigFile**

**specAccFileStr**

**isSkip**

**skipStatusMsgs**

**skipStatusMsgsRB**

**recruitYrStrt**

**recruitYrStop**

### Detailed Description

This class allows the user to adjust parameters used in computing scallop growth.

### Constructor & Destructor Documentation

#### GrowthFrame.Growth.\_\_init\_\_ ( *self*, *container*, *friend*)

Constructor for Growth Class.

Reimplemented from **GeoSams.MainApplication** (*p.78*).

### Member Function Documentation

#### GrowthFrame.Growth.GetGrowthConfigFName ( *self*)

Calls the filedialog method asksaveasfilename to name a file to be used for the Mortality Configuration file.

It then writes out the defined parameters to this file using the 'tag = value' format.

#### GrowthFrame.Growth.LoadGrowthData ( *self*)

#### GrowthFrame.Growth.on\_visibility ( *self*, *event*)

#### GrowthFrame.Growth.pop\_up ( *self*)

Reimplemented from **GeoSams.MainApplication** (*p.79*).

#### GrowthFrame.Growth.UpdateValues ( *self*)

Method to read Mortality Configuration file and set values accordingly.

#### GrowthFrame.Growth.UpdateWidgets ( *self*)

### Member Data Documentation

#### GrowthFrame.Growth.alphaMort

#### GrowthFrame.Growth.alphStr

#### GrowthFrame.Growth.dredgeWdStr

#### GrowthFrame.Growth.dredgeWth

#### GrowthFrame.Growth.fishMort

#### GrowthFrame.Growth.fmortFileName

#### GrowthFrame.Growth.fmortStr

#### GrowthFrame.Growth.friend

#### GrowthFrame.Growth.gbAdultMort

#### GrowthFrame.Growth.gbAdultMortStr

#### GrowthFrame.Growth.gbClFSelAStr

#### GrowthFrame.Growth.gbClFSelBStr

#### GrowthFrame.Growth.gbClosedFSelectA

#### GrowthFrame.Growth.gbClosedFSelectB

#### GrowthFrame.Growth.gbCullSize

#### GrowthFrame.Growth.gbCullStr

#### GrowthFrame.Growth.gbDiscard

#### GrowthFrame.Growth.gbDiscStr

#### GrowthFrame.Growth.gbIncident

#### GrowthFrame.Growth.gbIncidStr

#### GrowthFrame.Growth.gbLen0Str

#### GrowthFrame.Growth.gbLength0

#### GrowthFrame.Growth.gbOpenFSelectA

#### GrowthFrame.Growth.gbOpenFSelectB

#### GrowthFrame.Growth.gbOpFSelAStr

#### GrowthFrame.Growth.gbOpFSelBStr

#### GrowthFrame.Growth.growthCfgFile

#### GrowthFrame.Growth.growthStartDir

#### GrowthFrame.Growth.loadGrowthConfigButton

#### GrowthFrame.Growth.lpueIntcept

#### GrowthFrame.Growth.lpueIntcStr

#### GrowthFrame.Growth.lpueSl2Str

#### GrowthFrame.Growth.lpueSlope

#### GrowthFrame.Growth.lpueSlope2

#### GrowthFrame.Growth.lpueSlStr

#### GrowthFrame.Growth.maAdultMort

#### GrowthFrame.Growth.maAdultMortStr

#### GrowthFrame.Growth.maCullSize

#### GrowthFrame.Growth.maCullStr

#### GrowthFrame.Growth.maDiscard

#### GrowthFrame.Growth.maDiscStr

#### GrowthFrame.Growth.maFSelAStr

#### GrowthFrame.Growth.maFSelBStr

#### GrowthFrame.Growth.maFSelectA

#### GrowthFrame.Growth.maFSelectB

#### GrowthFrame.Growth.maIncident

#### GrowthFrame.Growth.maIncidStr

#### GrowthFrame.Growth.maLen0Str

#### GrowthFrame.Growth.maLength0

#### GrowthFrame.Growth.maxPerDay

#### GrowthFrame.Growth.maxPerDayStr

#### GrowthFrame.Growth.maxTime

#### GrowthFrame.Growth.maxTimeStr

#### GrowthFrame.Growth.on\_visibility

#### GrowthFrame.Growth.root

#### GrowthFrame.Growth.saveGrowthConfigButton

#### GrowthFrame.Growth.towSpdStr

#### GrowthFrame.Growth.towSpeed

#### The documentation for this class was generated from the following file:

#### GrowthFrame.py

## GeoSams.MainApplication Class Reference

This class is the parent class for the GUI.

Inheritance diagram for GeoSams.MainApplication:

### Public Member Functions

**\_\_init\_\_** (self, title, **maxAreas**, **maxCorners**, **maxYears**)

**ToggleSkipStatusMsgs** (self)

**ShowArgs** (self)

*Display setup limits here Messagebox blocks entry widgets if attempted to open before the main window completes.*

**Run\_Sim** (self)

*Starts the GeoSAMS simulatation* ***ScallopPopDensity*** *.*

**InterpAndPlotResults** (self)

**SaveConfigFiles** (self)

*Save all of the defined configuration files.*

**WriteScallopConfig** (self)

*Saves simulation configuration file.*

**ConvertMonthDayToDayOfYr** (self, monthDayStr)

*This method is used to converty the recruitment start and stop dates from a string month numerical day into days in a year.*

**ConvertDayOfYrToMonthDay** (self, dayOfYear)

**ReadRecruitConfigFile** (self)

*Read in the (tag, value) parameters from the recruitment to update parameters.*

**WriteRecruitmentConfig** (self)

*Saves recruitment parameters to a configuration file.*

**WriteGrowthConfig** (self)

*Saves mortality parameters to a configuration file.*

**WriteGridMgrConfig** (self)

*Saves grid manager parameters to a configuration file.*

**WriteSpatialFncsConfig** (self, cfgFile, functions, numFncsEntry)

*Saves spatial function parameters to a configuration file.*

**ReadConfigFile** (self, fName)

*Reads a typical configuration file to recover the tags and values.*

**ReadSimConfigFile** (self)

*Read in the (tag, value) parameters from the simulation configuration file.*

**ReadGridMgrConfigFile** (self)

*Read in the (tag, value) parameters from the grid manager configuration file.*

**pop\_up** (self)

### Public Attributes

**monDictdaysInYear**

**maxAreas**

**maxCorners**

**maxYears**

**addFrameClicked**

**tsPerYear**

**paramVal**

**paramStr**

**useHabCamData**

**domainName**

**yearStart**

**yearStop**

**simConfigFile**

**style**

**root**

**notebook**

**frame1**

**recrConfigFile**

**frame2**

**frame3**

**frame5**

**frame6**

**frame7**

**frame8**

**gmConfigFile**

**specAccFileStr**

**isSkip**

**skipStatusMsgs**

**skipStatusMsgsRB**

**recruitYrStrt**

**recruitYrStop**

### Detailed Description

This class is the parent class for the GUI.

### Constructor & Destructor Documentation

#### GeoSams.MainApplication.\_\_init\_\_ ( *self*, *title*, *maxAreas*, *maxCorners*, *maxYears*)

Reimplemented in **GrowthFrame.Growth** (*p.71*).

### Member Function Documentation

#### GeoSams.MainApplication.ConvertDayOfYrToMonthDay ( *self*, *dayOfYear*)

#### GeoSams.MainApplication.ConvertMonthDayToDayOfYr ( *self*, *monthDayStr*)

This method is used to converty the recruitment start and stop dates from a string month numerical day into days in a year.

Changed entry to combo box to guarantee format

##### Parameters

|  |  |
| --- | --- |
| *monthDayStr* | string that holds month and day in alpha format. That is 'JAN 01' |

The Growth year starts on June 1st, actually May 31 at 2400 Jun 1st @ 0600 is day 0.25 which is = 0.25 /365.2425 = 0.00068 years June 1st @ 1200 is day 0.50 which is = 0.50 /365.2425 = 0.00137

June 1st @ 1800 is day 0.75 which is = 0.75 /365.2425 = 0.00205

June 1st @ 2359 is day 0.99 which is = 0.99931/365.2425 = 0.002736

Jun2 2nd @ 0000 is day 1 which is = 1.00000/365.2425 = 0.00274

Jun2 2nd @ 2400 is day 2 which is = 2.00000/365.2425 = 0.00548

Dec 31st @ 2400 is day 214 which is = 214. /365.2425 = 0.58591

Jan 1st @ 2400 is day 215 which is = 215. /365.2425 = 0.58865

= 1 + DayOfYear(12,31) - DayOfYear(5,31) Apr 10 @ 2400 is day 314 which is = 314. /365.2425 = 0.85970

if leap year 315 which is = 315. /365.2425 = 0.86244 However, leap year is handled in the main loop in which it is considered only for the current year

#### GeoSams.MainApplication.InterpAndPlotResults ( *self*)

#### GeoSams.MainApplication.pop\_up ( *self*)

Reimplemented in **GrowthFrame.Growth** (*p.71*).

#### GeoSams.MainApplication.ReadConfigFile ( *self*, *fName*)

Reads a typical configuration file to recover the tags and values.

The parameters in these files all have the following format:

# indicates that the line is a comment. Otherwise

'tag' = 'value

##### Parameters

|  |  |
| --- | --- |
| *fName* | The name of the file to read. |

##### Returns

An array of tuples showing (tag, value) found in the file

#### GeoSams.MainApplication.ReadGridMgrConfigFile ( *self*)

Read in the (tag, value) parameters from the grid manager configuration file.

#### GeoSams.MainApplication.ReadRecruitConfigFile ( *self*)

Read in the (tag, value) parameters from the recruitment to update parameters.

#### GeoSams.MainApplication.ReadSimConfigFile ( *self*)

Read in the (tag, value) parameters from the simulation configuration file.

#### GeoSams.MainApplication.Run\_Sim ( *self*)

Starts the GeoSAMS simulatation **ScallopPopDensity** .

1a) TrawlData5mmbin(Year, 'DN') this will Delete bin5mm2022AL.csv INPUT: OriginalData\dredgetowbysize7917.csv OUTPUT: Data\bin5mm<yyyy><dn>.csv

#### GeoSams.MainApplication.SaveConfigFiles ( *self*)

Save all of the defined configuration files.

#### GeoSams.MainApplication.ShowArgs ( *self*)

Display setup limits here Messagebox blocks entry widgets if attempted to open before the main window completes.

#### GeoSams.MainApplication.ToggleSkipStatusMsgs ( *self*)

#### GeoSams.MainApplication.WriteGridMgrConfig ( *self*)

Saves grid manager parameters to a configuration file.

#### GeoSams.MainApplication.WriteGrowthConfig ( *self*)

Saves mortality parameters to a configuration file.

#### GeoSams.MainApplication.WriteRecruitmentConfig ( *self*)

Saves recruitment parameters to a configuration file.

#### GeoSams.MainApplication.WriteScallopConfig ( *self*)

Saves simulation configuration file.

It does so by writeing the parameters for the to the name file as well as keeping helpfule comments.

#### GeoSams.MainApplication.WriteSpatialFncsConfig ( *self*, *cfgFile*, *functions*, *numFncsEntry*)

Saves spatial function parameters to a configuration file.

### Member Data Documentation

#### GeoSams.MainApplication.addFrameClicked

#### GeoSams.MainApplication.daysInYear

#### GeoSams.MainApplication.domainName

#### GeoSams.MainApplication.frame1

#### GeoSams.MainApplication.frame2

#### GeoSams.MainApplication.frame3

#### GeoSams.MainApplication.frame5

#### GeoSams.MainApplication.frame6

#### GeoSams.MainApplication.frame7

#### GeoSams.MainApplication.frame8

#### GeoSams.MainApplication.gmConfigFile

#### GeoSams.MainApplication.isSkip

#### GeoSams.MainApplication.maxAreas

#### GeoSams.MainApplication.maxCorners

#### GeoSams.MainApplication.maxYears

#### GeoSams.MainApplication.monDict

#### GeoSams.MainApplication.notebook

#### GeoSams.MainApplication.paramStr

#### GeoSams.MainApplication.paramVal

#### GeoSams.MainApplication.recrConfigFile

#### GeoSams.MainApplication.recruitYrStop

#### GeoSams.MainApplication.recruitYrStrt

#### GeoSams.MainApplication.root

#### GeoSams.MainApplication.simConfigFile

#### GeoSams.MainApplication.skipStatusMsgs

#### GeoSams.MainApplication.skipStatusMsgsRB

#### GeoSams.MainApplication.specAccFileStr

#### GeoSams.MainApplication.style

#### GeoSams.MainApplication.tsPerYear

#### GeoSams.MainApplication.useHabCamData

#### GeoSams.MainApplication.yearStart

#### GeoSams.MainApplication.yearStop

#### The documentation for this class was generated from the following file:

#### GeoSams.py

## MainInputFrame.MainInput Class Reference

This class displays information about GeoSAMS simulation.

Inheritance diagram for MainInputFrame.MainInput:

### Public Member Functions

**\_\_init\_\_** (self, container, **friend**, **tsPerYear**, selectedOutputs, **maxYears**)

**EnterKeyClicked** (self, event)

*This method is called on both Enter Key clicked and goes out of focus.*

**CheckStartDay** (self, event)

*Checks start day to validate date is appropriate for month.*

**CheckStopDay** (self, event)

*Checks stop day to validate date is appropriate for month.*

**CBSelectedOutput** (self)

*Determines the value for which outputs are selected as they are checked.*

**GetSelectedOutputs** (self)

*Updates the final value from which outputs are selected.*

**ComputeSelectOuputValue** (self)

*Bit shifts (multiplies) checkbuttons and computes bit position value.*

**GetRecrConfigFName** (self)

*Calls the filedialog method asksaveasfilename to name a file to be used for the Recruitment Configuration file.*

**GetSimConfigFName** (self)

*Calls the filedialog method asksaveasfilename to name a file to be used for the Simulation Configuration file.*

**SetDredgeFileName** (self)

**SetDredgeFileEnvVar** (self)

**SetHabCamFileName** (self)

**SetHabCamFileEnvVar** (self)

**OpenPDF** (self)

**SetMaShapeFile** (self)

**SetMaShapeFileEnvVar** (self)

**SetGbShapeFile** (self)

**SetGBShapeFileEnvVar** (self)

**SetMaShapeBufferFile** (self)

**SetMaShapeBufferFileEnvVar** (self)

**SetGbShapeBufferFile** (self)

**SetGBShapeBufferFileEnvVar** (self)

**pop\_up** (self)

### Public Attributes

**rootsimStartDir**

**interpStartDir**

**surveyStartDir**

**shapeFileDir**

**friend**

**maxYears**

**recrCfgFile**

**simCfgFile**

**openRecrConfigButton**

**openSimConfigButton**

**openPDFButton**

**dredgeDataFile**

**habCamDataFile**

**setDredgeDataButton**

**setHabCamDataButton**

**maShapeFileEntry**

**gbShapeFileEntry**

**maShapeBufferFileEntry**

**gbShapeBufferFileEntry**

**setMaShapeFileButton**

**setGbShapeFileButton**

**setMaShapeBufferFileButton**

**setGbShapeBufferFileButton**

**monthsArr**

**startDayLabel**

**startDayComboMonth**

**CheckStartDay**

**startDayComboDay**

**stopDayLabel**

**stopDayComboMonth**

**CheckStopDay**

**stopDayComboDay**

**recrYrStrt**

**recrYrStop**

**numYrsAvg**

**startYr**

**stopYr**

**tsPerYear**

**domainNameLabel**

**domainNameCombo**

**lpueVar**

**ebmsVar**

**bmsVar**

**abunVar**

**lndwVar**

**landVar**

**feffVar**

**fmortVar**

**recrVar**

**desiredOutput**

**areaFName**

### Detailed Description

This class displays information about GeoSAMS simulation.

This same information is used on the command line when starting SRC\ScallopPopDensity

### Constructor & Destructor Documentation

#### MainInputFrame.MainInput.\_\_init\_\_ ( *self*, *container*, *friend*, *tsPerYear*, *selectedOutputs*, *maxYears*)

### Member Function Documentation

#### MainInputFrame.MainInput.CBSelectedOutput ( *self*)

Determines the value for which outputs are selected as they are checked.

#### MainInputFrame.MainInput.CheckStartDay ( *self*, *event*)

Checks start day to validate date is appropriate for month.

Does not consider if leap year

#### MainInputFrame.MainInput.CheckStopDay ( *self*, *event*)

Checks stop day to validate date is appropriate for month.

Does not consider if leap year

#### MainInputFrame.MainInput.ComputeSelectOuputValue ( *self*)

Bit shifts (multiplies) checkbuttons and computes bit position value.

#### MainInputFrame.MainInput.EnterKeyClicked ( *self*, *event*)

This method is called on both Enter Key clicked and goes out of focus.

#### MainInputFrame.MainInput.GetRecrConfigFName ( *self*)

Calls the filedialog method asksaveasfilename to name a file to be used for the Recruitment Configuration file.

It then writes out the defined parameters to this file using the 'tag = value' format.

#### MainInputFrame.MainInput.GetSelectedOutputs ( *self*)

Updates the final value from which outputs are selected.

#### MainInputFrame.MainInput.GetSimConfigFName ( *self*)

Calls the filedialog method asksaveasfilename to name a file to be used for the Simulation Configuration file.

It then writes out the defined parameters to this file using the 'tag = value' format.

#### MainInputFrame.MainInput.OpenPDF ( *self*)

#### MainInputFrame.MainInput.pop\_up ( *self*)

#### MainInputFrame.MainInput.SetDredgeFileEnvVar ( *self*)

#### MainInputFrame.MainInput.SetDredgeFileName ( *self*)

#### MainInputFrame.MainInput.SetGbShapeBufferFile ( *self*)

#### MainInputFrame.MainInput.SetGBShapeBufferFileEnvVar ( *self*)

#### MainInputFrame.MainInput.SetGbShapeFile ( *self*)

#### MainInputFrame.MainInput.SetGBShapeFileEnvVar ( *self*)

#### MainInputFrame.MainInput.SetHabCamFileEnvVar ( *self*)

#### MainInputFrame.MainInput.SetHabCamFileName ( *self*)

#### MainInputFrame.MainInput.SetMaShapeBufferFile ( *self*)

#### MainInputFrame.MainInput.SetMaShapeBufferFileEnvVar ( *self*)

#### MainInputFrame.MainInput.SetMaShapeFile ( *self*)

#### MainInputFrame.MainInput.SetMaShapeFileEnvVar ( *self*)

### Member Data Documentation

#### MainInputFrame.MainInput.abunVar

#### MainInputFrame.MainInput.areaFName

#### MainInputFrame.MainInput.bmsVar

#### MainInputFrame.MainInput.CheckStartDay

#### MainInputFrame.MainInput.CheckStopDay

#### MainInputFrame.MainInput.desiredOutput

#### MainInputFrame.MainInput.domainNameCombo

#### MainInputFrame.MainInput.domainNameLabel

#### MainInputFrame.MainInput.dredgeDataFile

#### MainInputFrame.MainInput.ebmsVar

#### MainInputFrame.MainInput.feffVar

#### MainInputFrame.MainInput.fmortVar

#### MainInputFrame.MainInput.friend

#### MainInputFrame.MainInput.gbShapeBufferFileEntry

#### MainInputFrame.MainInput.gbShapeFileEntry

#### MainInputFrame.MainInput.habCamDataFile

#### MainInputFrame.MainInput.interpStartDir

#### MainInputFrame.MainInput.landVar

#### MainInputFrame.MainInput.lndwVar

#### MainInputFrame.MainInput.lpueVar

#### MainInputFrame.MainInput.maShapeBufferFileEntry

#### MainInputFrame.MainInput.maShapeFileEntry

#### MainInputFrame.MainInput.maxYears

#### MainInputFrame.MainInput.monthsArr

#### MainInputFrame.MainInput.numYrsAvg

#### MainInputFrame.MainInput.openPDFButton

#### MainInputFrame.MainInput.openRecrConfigButton

#### MainInputFrame.MainInput.openSimConfigButton

#### MainInputFrame.MainInput.recrCfgFile

#### MainInputFrame.MainInput.recrVar

#### MainInputFrame.MainInput.recrYrStop

#### MainInputFrame.MainInput.recrYrStrt

#### MainInputFrame.MainInput.root

#### MainInputFrame.MainInput.setDredgeDataButton

#### MainInputFrame.MainInput.setGbShapeBufferFileButton

#### MainInputFrame.MainInput.setGbShapeFileButton

#### MainInputFrame.MainInput.setHabCamDataButton

#### MainInputFrame.MainInput.setMaShapeBufferFileButton

#### MainInputFrame.MainInput.setMaShapeFileButton

#### MainInputFrame.MainInput.shapeFileDir

#### MainInputFrame.MainInput.simCfgFile

#### MainInputFrame.MainInput.simStartDir

#### MainInputFrame.MainInput.startDayComboDay

#### MainInputFrame.MainInput.startDayComboMonth

#### MainInputFrame.MainInput.startDayLabel

#### MainInputFrame.MainInput.startYr

#### MainInputFrame.MainInput.stopDayComboDay

#### MainInputFrame.MainInput.stopDayComboMonth

#### MainInputFrame.MainInput.stopDayLabel

#### MainInputFrame.MainInput.stopYr

#### MainInputFrame.MainInput.surveyStartDir

#### MainInputFrame.MainInput.tsPerYear

#### The documentation for this class was generated from the following file:

#### MainInputFrame.py

## shapefile.Reader Class Reference

Inheritance diagram for shapefile.Reader:

### Public Member Functions

**\_\_init\_\_** (self, \*args, \*\*kwargs)

**\_\_str\_\_** (self)

**\_\_enter\_\_** (self)

**\_\_exit\_\_** (self, exc\_type, exc\_val, exc\_tb)

**\_\_len\_\_** (self)

**\_\_iter\_\_** (self)

**\_\_geo\_interface\_\_** (self)

**shapeTypeName** (self)

**load** (self, shapefile=None)

**load\_shp** (self, shapefile\_name)

**load\_shx** (self, shapefile\_name)

**load\_dbf** (self, shapefile\_name)

**\_\_del\_\_** (self)

**close** (self)

**shape** (self, i=0, **bbox**=None)

**shapes** (self, **bbox**=None)

**iterShapes** (self, **bbox**=None)

**record** (self, i=0, **fields**=None)

**records** (self, **fields**=None)

**iterRecords** (self, **fields**=None)

**shapeRecord** (self, i=0, **fields**=None, **bbox**=None)

**shapeRecords** (self, **fields**=None, **bbox**=None)

**iterShapeRecords** (self, **fields**=None, **bbox**=None)

### Public Attributes

**shpshx**

**dbf**

**shapeName**

**shpLength**

**numRecords**

**numShapes**

**fields**

**encoding**

**encodingErrors**

**shapeType**

**bbox**

**zbox**

**mbox**

### Protected Attributes

**\_files\_to\_close\_offsets**

### Private Member Functions

**\_\_getFileObj** (self, f)

**\_\_restrictIndex** (self, i)

**\_\_shpHeader** (self)

**\_\_shape** (self, oid=None, **bbox**=None)

**\_\_shxHeader** (self)

**\_\_shxOffsets** (self)

**\_\_shapeIndex** (self, i=None)

**\_\_dbfHeader** (self)

**\_\_recordFmt** (self, **fields**=None)

**\_\_recordFields** (self, **fields**=None)

**\_\_record** (self, fieldTuples, recLookup, recStruct, oid=None)

### Private Attributes

**\_\_dbfHdrLength\_\_fieldLookup**

**\_\_recordLength**

**\_\_fullRecStruct**

**\_\_fullRecLookup**

### Detailed Description

Reads the three files of a shapefile as a unit or

separately. If one of the three files (.shp, .shx,

.dbf) is missing no exception is thrown until you try

to call a method that depends on that particular file.

The .shx index file is used if available for efficiency

but is not required to read the geometry from the .shp

file. The "shapefile" argument in the constructor is the

name of the file you want to open, and can be the path

to a shapefile on a local filesystem, inside a zipfile,

or a url.

You can instantiate a Reader without specifying a shapefile

and then specify one later with the load() method.

Only the shapefile headers are read upon loading. Content

within each file is only accessed when required and as

efficiently as possible. Shapefiles are usually not large

but they can be.

### Constructor & Destructor Documentation

#### shapefile.Reader.\_\_init\_\_ ( *self*, \* *args*, \*\* *kwargs*)

#### shapefile.Reader.\_\_del\_\_ ( *self*)

### Member Function Documentation

#### shapefile.Reader.\_\_dbfHeader ( *self*)[private]

Reads a dbf header. Xbase-related code borrows heavily from ActiveState Python Cookbook Recipe 362715 by Raymond Hettinger

#### shapefile.Reader.\_\_enter\_\_ ( *self*)

Enter phase of context manager.

#### shapefile.Reader.\_\_exit\_\_ ( *self*, *exc\_type*, *exc\_val*, *exc\_tb*)

Exit phase of context manager, close opened files.

#### shapefile.Reader.\_\_geo\_interface\_\_ ( *self*)

#### shapefile.Reader.\_\_getFileObj ( *self*, *f*)[private]

Checks to see if the requested shapefile file object is

available. If not a ShapefileException is raised.

#### shapefile.Reader.\_\_iter\_\_ ( *self*)

Iterates through the shapes/records in the shapefile.

#### shapefile.Reader.\_\_len\_\_ ( *self*)

Returns the number of shapes/records in the shapefile.

#### shapefile.Reader.\_\_record ( *self*, *fieldTuples*, *recLookup*, *recStruct*, *oid* = None)[private]

Reads and returns a dbf record row as a list of values. Requires specifying

a list of field info tuples 'fieldTuples', a record name-index dict 'recLookup',

and a Struct instance 'recStruct' for unpacking these fields.

#### shapefile.Reader.\_\_recordFields ( *self*, *fields* = None)[private]

Returns the necessary info required to unpack a record's fields,

restricted to a subset of fieldnames 'fields' if specified.

Returns a list of field info tuples, a name-index lookup dict,

and a Struct instance for unpacking these fields. Note that DeletionFlag

is not a valid field.

#### shapefile.Reader.\_\_recordFmt ( *self*, *fields* = None)[private]

Calculates the format and size of a .dbf record. Optional 'fields' arg

specifies which fieldnames to unpack and which to ignore. Note that this

always includes the DeletionFlag at index 0, regardless of the 'fields' arg.

#### shapefile.Reader.\_\_restrictIndex ( *self*, *i*)[private]

Provides list-like handling of a record index with a clearer

error message if the index is out of bounds.

#### shapefile.Reader.\_\_shape ( *self*, *oid* = None, *bbox* = None)[private]

Returns the header info and geometry for a single shape.

#### shapefile.Reader.\_\_shapeIndex ( *self*, *i* = None)[private]

Returns the offset in a .shp file for a shape based on information

in the .shx index file.

#### shapefile.Reader.\_\_shpHeader ( *self*)[private]

Reads the header information from a .shp file.

#### shapefile.Reader.\_\_shxHeader ( *self*)[private]

Reads the header information from a .shx file.

#### shapefile.Reader.\_\_shxOffsets ( *self*)[private]

Reads the shape offset positions from a .shx file

#### shapefile.Reader.\_\_str\_\_ ( *self*)

Use some general info on the shapefile as \_\_str\_\_

#### shapefile.Reader.close ( *self*)

#### shapefile.Reader.iterRecords ( *self*, *fields* = None)

Returns a generator of records in a dbf file.

Useful for large shapefiles or dbf files.

To only read some of the fields, specify the 'fields' arg as a

list of one or more fieldnames.

#### shapefile.Reader.iterShapeRecords ( *self*, *fields* = None, *bbox* = None)

Returns a generator of combination geometry/attribute records for

all records in a shapefile.

To only read some of the fields, specify the 'fields' arg as a

list of one or more fieldnames.

To only read entries within a given spatial region, specify the 'bbox'

arg as a list or tuple of xmin,ymin,xmax,ymax.

#### shapefile.Reader.iterShapes ( *self*, *bbox* = None)

Returns a generator of shapes in a shapefile. Useful

for handling large shapefiles.

To only read shapes within a given spatial region, specify the 'bbox'

arg as a list or tuple of xmin,ymin,xmax,ymax.

#### shapefile.Reader.load ( *self*, *shapefile* = None)

Opens a shapefile from a filename or file-like

object. Normally this method would be called by the

constructor with the file name as an argument.

#### shapefile.Reader.load\_dbf ( *self*, *shapefile\_name*)

Attempts to load file with .dbf extension as both lower and upper case

#### shapefile.Reader.load\_shp ( *self*, *shapefile\_name*)

Attempts to load file with .shp extension as both lower and upper case

#### shapefile.Reader.load\_shx ( *self*, *shapefile\_name*)

Attempts to load file with .shx extension as both lower and upper case

#### shapefile.Reader.record ( *self*, *i* = 0, *fields* = None)

Returns a specific dbf record based on the supplied index.

To only read some of the fields, specify the 'fields' arg as a

list of one or more fieldnames.

#### shapefile.Reader.records ( *self*, *fields* = None)

Returns all records in a dbf file.

To only read some of the fields, specify the 'fields' arg as a

list of one or more fieldnames.

#### shapefile.Reader.shape ( *self*, *i* = 0, *bbox* = None)

Returns a shape object for a shape in the geometry

record file.

If the 'bbox' arg is given (list or tuple of xmin,ymin,xmax,ymax),

returns None if the shape is not within that region.

#### shapefile.Reader.shapeRecord ( *self*, *i* = 0, *fields* = None, *bbox* = None)

Returns a combination geometry and attribute record for the

supplied record index.

To only read some of the fields, specify the 'fields' arg as a

list of one or more fieldnames.

If the 'bbox' arg is given (list or tuple of xmin,ymin,xmax,ymax),

returns None if the shape is not within that region.

#### shapefile.Reader.shapeRecords ( *self*, *fields* = None, *bbox* = None)

Returns a list of combination geometry/attribute records for

all records in a shapefile.

To only read some of the fields, specify the 'fields' arg as a

list of one or more fieldnames.

To only read entries within a given spatial region, specify the 'bbox'

arg as a list or tuple of xmin,ymin,xmax,ymax.

#### shapefile.Reader.shapes ( *self*, *bbox* = None)

Returns all shapes in a shapefile.

To only read shapes within a given spatial region, specify the 'bbox'

arg as a list or tuple of xmin,ymin,xmax,ymax.

#### shapefile.Reader.shapeTypeName ( *self*)

### Member Data Documentation

#### shapefile.Reader.\_\_dbfHdrLength[private]

#### shapefile.Reader.\_\_fieldLookup[private]

#### shapefile.Reader.\_\_fullRecLookup[private]

#### shapefile.Reader.\_\_fullRecStruct[private]

#### shapefile.Reader.\_\_recordLength[private]

#### shapefile.Reader.\_files\_to\_close[protected]

#### shapefile.Reader.\_offsets[protected]

#### shapefile.Reader.bbox

#### shapefile.Reader.dbf

#### shapefile.Reader.encoding

#### shapefile.Reader.encodingErrors

#### shapefile.Reader.fields

#### shapefile.Reader.mbox

#### shapefile.Reader.numRecords

#### shapefile.Reader.numShapes

#### shapefile.Reader.shapeName

#### shapefile.Reader.shapeType

#### shapefile.Reader.shp

#### shapefile.Reader.shpLength

#### shapefile.Reader.shx

#### shapefile.Reader.zbox

#### The documentation for this class was generated from the following file:

PyshpMaster/**shapefile.py**

## Widgets.ScrollFrame Class Reference

Scrollable Frame Class from https://gist.github.com/mp035/9f2027c3ef9172264532fcd6262f3b01.

Inheritance diagram for Widgets.ScrollFrame:

### Public Member Functions

**\_\_init\_\_** (self, parent)

**onFrameConfigure** (self, event)

*whenever the size of the frame changes, alter the scroll region respectively.*

**onMouseWheel** (self, event)

*cross platform scroll wheel event*

**onEnter** (self, event)

*bind wheel events when the cursor enters the control*

**onLeave** (self, event)

*unbind wheel events when the cursorl leaves the control*

### Public Attributes

**canvasviewPort**

**canvas\_window**

**onFrameConfigure**

**onEnter**

**onLeave**

**onMouseWheel**

### Detailed Description

Scrollable Frame Class from https://gist.github.com/mp035/9f2027c3ef9172264532fcd6262f3b01.

### Constructor & Destructor Documentation

#### Widgets.ScrollFrame.\_\_init\_\_ ( *self*, *parent*)

### Member Function Documentation

#### Widgets.ScrollFrame.onEnter ( *self*, *event*)

bind wheel events when the cursor enters the control

#### Widgets.ScrollFrame.onFrameConfigure ( *self*, *event*)

whenever the size of the frame changes, alter the scroll region respectively.

Reset the scroll region to encompass the inner frame

#### Widgets.ScrollFrame.onLeave ( *self*, *event*)

unbind wheel events when the cursorl leaves the control

#### Widgets.ScrollFrame.onMouseWheel ( *self*, *event*)

cross platform scroll wheel event

### Member Data Documentation

#### Widgets.ScrollFrame.canvas

#### Widgets.ScrollFrame.canvas\_window

#### Widgets.ScrollFrame.onEnter

#### Widgets.ScrollFrame.onFrameConfigure

#### Widgets.ScrollFrame.onLeave

#### Widgets.ScrollFrame.onMouseWheel

#### Widgets.ScrollFrame.viewPort

#### The documentation for this class was generated from the following file:

#### Widgets.py

## shapefile.Shape Class Reference

Inheritance diagram for shapefile.Shape:

### Public Member Functions

**\_\_init\_\_** (self, **shapeType**=**NULL**, **points**=None, **parts**=None, **partTypes**=None, **oid**=None)

**\_\_geo\_interface\_\_** (self)

**oid** (self)

**shapeTypeName** (self)

**\_\_repr\_\_** (self)

### Public Attributes

**shapeTypepoints**

**parts**

**partTypes**

**shapeTypeName**

### Static Protected Member Functions

**\_from\_geojson** (geoj)

### Protected Attributes

### \_errorsPrivate Attributes

**\_\_oid**

### Constructor & Destructor Documentation

#### shapefile.Shape.\_\_init\_\_ ( *self*, *shapeType* = NULL, *points* = None, *parts* = None, *partTypes* = None, *oid* = None)

Stores the geometry of the different shape types

specified in the Shapefile spec. Shape types are

usually point, polyline, or polygons. Every shape type

except the "Null" type contains points at some level for

example vertices in a polygon. If a shape type has

multiple shapes containing points within a single

geometry record then those shapes are called parts. Parts

are designated by their starting index in geometry record's

list of shapes. For MultiPatch geometry, partTypes designates

the patch type of each of the parts.

### Member Function Documentation

#### shapefile.Shape.\_\_geo\_interface\_\_ ( *self*)

#### shapefile.Shape.\_\_repr\_\_ ( *self*)

#### shapefile.Shape.\_from\_geojson ( *geoj*)[static], [protected]

#### shapefile.Shape.oid ( *self*)

The index position of the shape in the original shapefile

#### shapefile.Shape.shapeTypeName ( *self*)

### Member Data Documentation

#### shapefile.Shape.\_\_oid[private]

#### shapefile.Shape.\_errors[protected]

#### shapefile.Shape.parts

#### shapefile.Shape.partTypes

#### shapefile.Shape.points

#### shapefile.Shape.shapeType

#### shapefile.Shape.shapeTypeName

#### The documentation for this class was generated from the following file:

PyshpMaster/**shapefile.py**

## shapefile.ShapefileException Class Reference

Inheritance diagram for shapefile.ShapefileException:

### Detailed Description

An exception to handle shapefile specific problems.

The documentation for this class was generated from the following file:

PyshpMaster/**shapefile.py**

## shapefile.ShapeRecord Class Reference

Inheritance diagram for shapefile.ShapeRecord:

### Public Member Functions

**\_\_init\_\_** (self, **shape**=None, **record**=None)

**\_\_geo\_interface\_\_** (self)

### Public Attributes

**shaperecord**

### Detailed Description

A ShapeRecord object containing a shape along with its attributes.

Provides the GeoJSON \_\_geo\_interface\_\_ to return a Feature dictionary.

### Constructor & Destructor Documentation

#### shapefile.ShapeRecord.\_\_init\_\_ ( *self*, *shape* = None, *record* = None)

### Member Function Documentation

#### shapefile.ShapeRecord.\_\_geo\_interface\_\_ ( *self*)

### Member Data Documentation

#### shapefile.ShapeRecord.record

#### shapefile.ShapeRecord.shape

#### The documentation for this class was generated from the following file:

PyshpMaster/**shapefile.py**

## shapefile.ShapeRecords Class Reference

Inheritance diagram for shapefile.ShapeRecords:

### Public Member Functions

**\_\_repr\_\_** (self)

**\_\_geo\_interface\_\_** (self)

### Detailed Description

A class to hold a list of ShapeRecord objects. Subclasses list to ensure compatibility with

former work and to reuse all the optimizations of the builtin list.

In addition to the list interface, this also provides the GeoJSON \_\_geo\_interface\_\_

to return a FeatureCollection dictionary.

### Member Function Documentation

#### shapefile.ShapeRecords.\_\_geo\_interface\_\_ ( *self*)

#### shapefile.ShapeRecords.\_\_repr\_\_ ( *self*)

#### The documentation for this class was generated from the following file:

PyshpMaster/**shapefile.py**

## shapefile.Shapes Class Reference

Inheritance diagram for shapefile.Shapes:

### Public Member Functions

**\_\_repr\_\_** (self)

**\_\_geo\_interface\_\_** (self)

### Detailed Description

A class to hold a list of Shape objects. Subclasses list to ensure compatibility with

former work and to reuse all the optimizations of the builtin list.

In addition to the list interface, this also provides the GeoJSON \_\_geo\_interface\_\_

to return a GeometryCollection dictionary.

### Member Function Documentation

#### shapefile.Shapes.\_\_geo\_interface\_\_ ( *self*)

#### shapefile.Shapes.\_\_repr\_\_ ( *self*)

#### The documentation for this class was generated from the following file:

PyshpMaster/**shapefile.py**

## SortByAreaFrame.SortByArea Class Reference

This class is used to assist the user in defining areas of interest to assess accumulated parameters located in these areas of interest.

Inheritance diagram for SortByAreaFrame.SortByArea:

### Public Member Functions

**\_\_init\_\_** (self, container, **friend**, maxAreas, maxCorners, **maxYears**, **paramStr**)

**on\_visibility** (self, event)

**CbUpdateUnits** (self, event)

**UpdateUnits** (self)

**AppendYears** (self, addYears)

**RunSort** (self)

**UpdateWidgets** (self)

**GetDataSortFile** (self)

**SaveDataSortFile** (self)

**BrowseExportFile** (self)

**ExportThis** (self, nomsg=False)

*This method exports the current page of data, just a single output parameter.*

**ExportAll** (self)

*Export all select parameters.*

**EnterKeyClicked** (self, event)

**NumAreasUpdate** (self)

**pop\_up** (self)

*Help Window for Sort By Area.*

### Public Attributes

**rootstartDir**

**exportFileName**

**friend**

**areaFName**

**numAreasMax**

**numCornersMax**

**maxYears**

**numAreas**

*f.write('AREA,YEAR,' + outStr + ' ('+ units + ')*

*') Write Header*

**numCornersparamStr**

**yearStart**

*f.write('AREA,YEAR,' + outStr + ' ('+ units + ')*

*') Write Header*

**yearStopnumYears**

*f.write('AREA,YEAR,' + outStr + ' ('+ units + ')*

*') Write Header*

**areaDatascrollFrame**

**numAreasLabel**

**numAreasEntry**

**EnterKeyClicked**

**outputParmLabel**

**comboParameter**

**CbUpdateUnits**

**dataSortFileLabel**

**dataSortFileEntry**

**dataSortUnitsLabel**

**dataSortUnitsEntry**

**openDataSortButton**

**saveDataSortButton**

**runSortButton**

**exportThisSortButton**

**exportAllSortButton**

**exportFileLabel**

**exportFileEntry**

**browseExportButton**

**areas**

**on\_visibility**

**domainName**

### Detailed Description

This class is used to assist the user in defining areas of interest to assess accumulated parameters located in these areas of interest.

##### Parameters

|  |  |
| --- | --- |
| *friend* | is used to access Start and Stop year from the Main Frame |
| *maxAreas* | defined at start up for the maximum allowed areas |
| *maxCorners* | defined at start up for the maximum allowed corners or nodes |
| *maxYears* | defined at start up for the maximum allowed year range |
| *paramStr* | defined at start up for the desired outputs |

### Constructor & Destructor Documentation

#### SortByAreaFrame.SortByArea.\_\_init\_\_ ( *self*, *container*, *friend*, *maxAreas*, *maxCorners*, *maxYears*, *paramStr*)

### Member Function Documentation

#### SortByAreaFrame.SortByArea.AppendYears ( *self*, *addYears*)

#### SortByAreaFrame.SortByArea.BrowseExportFile ( *self*)

#### SortByAreaFrame.SortByArea.CbUpdateUnits ( *self*, *event*)

#### SortByAreaFrame.SortByArea.EnterKeyClicked ( *self*, *event*)

#### SortByAreaFrame.SortByArea.ExportAll ( *self*)

Export all select parameters.

For each parameter

Verify data file exists, Lat\_Lon\_Grid\_ + ABUN\_ + AL + \_ 2015\_2017

#### SortByAreaFrame.SortByArea.ExportThis ( *self*, *nomsg* = False)

This method exports the current page of data, just a single output parameter.

First row : AREA YEAR PARAMETER (UNITS) 1 StartYear

1 ... 1 StopYear ... ... N StartYear

N ... N StopYear

#### SortByAreaFrame.SortByArea.GetDataSortFile ( *self*)

#### SortByAreaFrame.SortByArea.NumAreasUpdate ( *self*)

Updates the number of areas functions.

#### SortByAreaFrame.SortByArea.on\_visibility ( *self*, *event*)

#### SortByAreaFrame.SortByArea.pop\_up ( *self*)

Help Window for Sort By Area.

#### SortByAreaFrame.SortByArea.RunSort ( *self*)

#### SortByAreaFrame.SortByArea.SaveDataSortFile ( *self*)

#### SortByAreaFrame.SortByArea.UpdateUnits ( *self*)

#### SortByAreaFrame.SortByArea.UpdateWidgets ( *self*)

### Member Data Documentation

#### SortByAreaFrame.SortByArea.areaData

#### SortByAreaFrame.SortByArea.areaFName

#### SortByAreaFrame.SortByArea.areas

#### SortByAreaFrame.SortByArea.browseExportButton

#### SortByAreaFrame.SortByArea.CbUpdateUnits

#### SortByAreaFrame.SortByArea.comboParameter

#### SortByAreaFrame.SortByArea.dataSortFileEntry

#### SortByAreaFrame.SortByArea.dataSortFileLabel

#### SortByAreaFrame.SortByArea.dataSortUnitsEntry

#### SortByAreaFrame.SortByArea.dataSortUnitsLabel

#### SortByAreaFrame.SortByArea.domainName

#### SortByAreaFrame.SortByArea.EnterKeyClicked

#### SortByAreaFrame.SortByArea.exportAllSortButton

#### SortByAreaFrame.SortByArea.exportFileEntry

#### SortByAreaFrame.SortByArea.exportFileLabel

#### SortByAreaFrame.SortByArea.exportFileName

#### SortByAreaFrame.SortByArea.exportThisSortButton

#### SortByAreaFrame.SortByArea.friend

#### SortByAreaFrame.SortByArea.maxYears

#### SortByAreaFrame.SortByArea.numAreas

f.write('AREA,YEAR,' + outStr + ' ('+ units + ')

') Write Header

#### SortByAreaFrame.SortByArea.numAreasEntry

#### SortByAreaFrame.SortByArea.numAreasLabel

#### SortByAreaFrame.SortByArea.numAreasMax

#### SortByAreaFrame.SortByArea.numCorners

#### SortByAreaFrame.SortByArea.numCornersMax

#### SortByAreaFrame.SortByArea.numYears

f.write('AREA,YEAR,' + outStr + ' ('+ units + ')

') Write Header

#### SortByAreaFrame.SortByArea.on\_visibility

#### SortByAreaFrame.SortByArea.openDataSortButton

#### SortByAreaFrame.SortByArea.outputParmLabel

#### SortByAreaFrame.SortByArea.paramStr

#### SortByAreaFrame.SortByArea.root

#### SortByAreaFrame.SortByArea.runSortButton

#### SortByAreaFrame.SortByArea.saveDataSortButton

#### SortByAreaFrame.SortByArea.scrollFrame

#### SortByAreaFrame.SortByArea.startDir

#### SortByAreaFrame.SortByArea.yearStart

f.write('AREA,YEAR,' + outStr + ' ('+ units + ')

') Write Header

#### SortByAreaFrame.SortByArea.yearStop

#### The documentation for this class was generated from the following file:

#### SortByAreaFrame.py

## SortByRegionFrame.SortByRegion Class Reference

This class is used to assist the user in defining areas of interest to assess accumulated parameters located in these areas of interest.

Inheritance diagram for SortByRegionFrame.SortByRegion:

### Public Member Functions

**\_\_init\_\_** (self, container, **friend**, **maxAreas**, **maxYears**, **paramStr**)

**on\_visibility** (self, event)

**AppendYears** (self, **numYears**)

*Max Number of years has increased, need to add additional columns.*

**RunSort** (self, showMsg=True)

**UpdateWidgets** (self)

**BrowseExportFile** (self)

**ExportThis** (self, exportingAll=False, showMsg=True)

*This method exports ouput parameter table to its own file name.*

**ExportAll** (self)

*Export all select parameters.*

**GetZoneData** (self)

*Gets shape data and places it into a array of GeoShape.*

**DetermineMgmtAreaIndex** (self, region)

*determine Management Area Index*

**pop\_up** (self)

*Help Window for Sort By Area.*

### Public Attributes

**rootstartDir**

**resultsDir**

**exportFileName**

**friend**

**areaFName**

**zones**

**maxAreas**

**maxYears**

**numAreas**

**numCorners**

**paramStr**

**areaKm2**

**yearStart**

**yearStop**

**numYears**

**scrollFrame**

**sortAreaFrame**

**outputParmLabel**

**comboParameter**

**runSortButton**

**exportThisSortButton**

**exportAllSortButton**

**exportFileLabel**

**exportFileEntry**

**browseExportButton**

**tableRows**

**tableCols**

**firstYrCol**

**table**

**on\_visibility**

**domainName**

### Detailed Description

This class is used to assist the user in defining areas of interest to assess accumulated parameters located in these areas of interest.

##### Parameters

|  |  |
| --- | --- |
| *friend* | is used to access Start and Stop year from the Main Frame |
| *maxAreas* | defined at start up for the maximum allowed areas |
| *maxCorners* | defined at start up for the maximum allowed corners or nodes |
| *maxYears* | defined at start up for the maximum allowed year range |
| *paramStr* | defined at start up for the desired outputs |

### Constructor & Destructor Documentation

#### SortByRegionFrame.SortByRegion.\_\_init\_\_ ( *self*, *container*, *friend*, *maxAreas*, *maxYears*, *paramStr*)

### Member Function Documentation

#### SortByRegionFrame.SortByRegion.AppendYears ( *self*, *numYears*)

Max Number of years has increased, need to add additional columns.

#### SortByRegionFrame.SortByRegion.BrowseExportFile ( *self*)

#### SortByRegionFrame.SortByRegion.DetermineMgmtAreaIndex ( *self*, *region*)

determine Management Area Index

#### SortByRegionFrame.SortByRegion.ExportAll ( *self*)

Export all select parameters.

For each parameter

Verify data file exists, Lat\_Lon\_Grid\_ + ABUN\_ + AL + \_ 2015\_2017

#### SortByRegionFrame.SortByRegion.ExportThis ( *self*, *exportingAll* = False, *showMsg* = True)

This method exports ouput parameter table to its own file name.

#### SortByRegionFrame.SortByRegion.GetZoneData ( *self*)

Gets shape data and places it into a array of GeoShape.

MA regions, if selected, are first placed into array followed by GB regions, selected. Logic will always have either MA, GB, or both

#### SortByRegionFrame.SortByRegion.on\_visibility ( *self*, *event*)

#### SortByRegionFrame.SortByRegion.pop\_up ( *self*)

Help Window for Sort By Area.

#### SortByRegionFrame.SortByRegion.RunSort ( *self*, *showMsg* = True)

#### SortByRegionFrame.SortByRegion.UpdateWidgets ( *self*)

### Member Data Documentation

#### SortByRegionFrame.SortByRegion.areaFName

#### SortByRegionFrame.SortByRegion.areaKm2

#### SortByRegionFrame.SortByRegion.browseExportButton

#### SortByRegionFrame.SortByRegion.comboParameter

#### SortByRegionFrame.SortByRegion.domainName

#### SortByRegionFrame.SortByRegion.exportAllSortButton

#### SortByRegionFrame.SortByRegion.exportFileEntry

#### SortByRegionFrame.SortByRegion.exportFileLabel

#### SortByRegionFrame.SortByRegion.exportFileName

#### SortByRegionFrame.SortByRegion.exportThisSortButton

#### SortByRegionFrame.SortByRegion.firstYrCol

#### SortByRegionFrame.SortByRegion.friend

#### SortByRegionFrame.SortByRegion.maxAreas

#### SortByRegionFrame.SortByRegion.maxYears

#### SortByRegionFrame.SortByRegion.numAreas

#### SortByRegionFrame.SortByRegion.numCorners

#### SortByRegionFrame.SortByRegion.numYears

#### SortByRegionFrame.SortByRegion.on\_visibility

#### SortByRegionFrame.SortByRegion.outputParmLabel

#### SortByRegionFrame.SortByRegion.paramStr

#### SortByRegionFrame.SortByRegion.resultsDir

#### SortByRegionFrame.SortByRegion.root

#### SortByRegionFrame.SortByRegion.runSortButton

#### SortByRegionFrame.SortByRegion.scrollFrame

#### SortByRegionFrame.SortByRegion.sortAreaFrame

#### SortByRegionFrame.SortByRegion.startDir

#### SortByRegionFrame.SortByRegion.table

#### SortByRegionFrame.SortByRegion.tableCols

#### SortByRegionFrame.SortByRegion.tableRows

#### SortByRegionFrame.SortByRegion.yearStart

#### SortByRegionFrame.SortByRegion.yearStop

#### SortByRegionFrame.SortByRegion.zones

#### The documentation for this class was generated from the following file:

#### SortByRegionFrame.py

## SpecialAreaFrame.SpecialArea Class Reference

This class is used to assist the user in defining areas of interest to assess accumulated parameters located in these areas of interest.

Inheritance diagram for SpecialAreaFrame.SpecialArea:

### Public Member Functions

**\_\_init\_\_** (self, container, **friend**, maxAreas, maxCorners)

**on\_visibility** (self, event)

**EnterKeyClicked** (self, event)

**GetGMgrConfigFName** (self)

*Calls the filedialog method asksaveasfilename to name a file to be used for the Grid Manager Configuration file.*

**NumAreasUpdate** (self)

**UpdateWidgets** (self)

**GetAreaFile** (self)

**SaveAreaFile** (self)

**SaveSpecialAreaData** (self, fname, **numAreas**)

**pop\_up** (self)

*Help Window for Special Access Area.*

### Public Attributes

**friendroot**

**simStartDir**

**startDir**

**areaFName**

**numAreasMax**

**numCornersMax**

**numAreas**

**numCorners**

**numAreasLabel**

**numAreasEntry**

**EnterKeyClicked**

**gmCfgFile**

**openGmgrConfigButton**

**openAreaFileButton**

**specAccFile**

**specAccFileLabel**

**saveAreaFileButton**

**areaMgr**

**on\_visibility**

### Detailed Description

This class is used to assist the user in defining areas of interest to assess accumulated parameters located in these areas of interest.

### Constructor & Destructor Documentation

#### SpecialAreaFrame.SpecialArea.\_\_init\_\_ ( *self*, *container*, *friend*, *maxAreas*, *maxCorners*)

### Member Function Documentation

#### SpecialAreaFrame.SpecialArea.EnterKeyClicked ( *self*, *event*)

#### SpecialAreaFrame.SpecialArea.GetAreaFile ( *self*)

#### SpecialAreaFrame.SpecialArea.GetGMgrConfigFName ( *self*)

Calls the filedialog method asksaveasfilename to name a file to be used for the Grid Manager Configuration file.

It then writes out the defined parameters to this file using the 'tag = value' format.

#### SpecialAreaFrame.SpecialArea.NumAreasUpdate ( *self*)

#### SpecialAreaFrame.SpecialArea.on\_visibility ( *self*, *event*)

#### SpecialAreaFrame.SpecialArea.pop\_up ( *self*)

Help Window for Special Access Area.

#### SpecialAreaFrame.SpecialArea.SaveAreaFile ( *self*)

#### SpecialAreaFrame.SpecialArea.SaveSpecialAreaData ( *self*, *fname*, *numAreas*)

#### SpecialAreaFrame.SpecialArea.UpdateWidgets ( *self*)

### Member Data Documentation

#### SpecialAreaFrame.SpecialArea.areaFName

#### SpecialAreaFrame.SpecialArea.areaMgr

#### SpecialAreaFrame.SpecialArea.EnterKeyClicked

#### SpecialAreaFrame.SpecialArea.friend

#### SpecialAreaFrame.SpecialArea.gmCfgFile

#### SpecialAreaFrame.SpecialArea.numAreas

#### SpecialAreaFrame.SpecialArea.numAreasEntry

#### SpecialAreaFrame.SpecialArea.numAreasLabel

#### SpecialAreaFrame.SpecialArea.numAreasMax

#### SpecialAreaFrame.SpecialArea.numCorners

#### SpecialAreaFrame.SpecialArea.numCornersMax

#### SpecialAreaFrame.SpecialArea.on\_visibility

#### SpecialAreaFrame.SpecialArea.openAreaFileButton

#### SpecialAreaFrame.SpecialArea.openGmgrConfigButton

#### SpecialAreaFrame.SpecialArea.root

#### SpecialAreaFrame.SpecialArea.saveAreaFileButton

#### SpecialAreaFrame.SpecialArea.simStartDir

#### SpecialAreaFrame.SpecialArea.specAccFile

#### SpecialAreaFrame.SpecialArea.specAccFileLabel

#### SpecialAreaFrame.SpecialArea.startDir

#### The documentation for this class was generated from the following file:

#### SpecialAreaFrame.py

## Widgets.SubFrameElement Class Reference

Generic Element.

Inheritance diagram for Widgets.SubFrameElement:

### Public Member Functions

**\_\_init\_\_** (self, container, parent, label, value, elementRow, labelCol, entryCol, width=5, valCmd=None, enterCmd=None)

### Public Attributes

**myEntrymyLabel**

### Detailed Description

Generic Element.

Provides a label and an entery field. Optionally allows programmer to specify a method to validate entry and another method to respond to Enter Key.

### Constructor & Destructor Documentation

#### Widgets.SubFrameElement.\_\_init\_\_ ( *self*, *container*, *parent*, *label*, *value*, *elementRow*, *labelCol*, *entryCol*, *width* = 5, *valCmd* = None, *enterCmd* = None)

### Member Data Documentation

#### Widgets.SubFrameElement.myEntry

#### Widgets.SubFrameElement.myLabel

#### The documentation for this class was generated from the following file:

#### Widgets.py

## Widgets.SubFrameInterpFunction Class Reference

Inheritance diagram for Widgets.SubFrameInterpFunction:

### Public Member Functions

**\_\_init\_\_** (self, container, parent, funcNum, dim, shape, preconNum, elementRow, elementCol)

### Public Attributes

**funcFramedimVal**

**myDimRBx**

**myDimRBy**

**myDimRBz**

**shapeVal**

**myShapeG**

**myShapeL**

**myShapeS**

**myShapeC**

**preconLabel**

**preconEntry**

### Constructor & Destructor Documentation

#### Widgets.SubFrameInterpFunction.\_\_init\_\_ ( *self*, *container*, *parent*, *funcNum*, *dim*, *shape*, *preconNum*, *elementRow*, *elementCol*)

### Member Data Documentation

#### Widgets.SubFrameInterpFunction.dimVal

#### Widgets.SubFrameInterpFunction.funcFrame

#### Widgets.SubFrameInterpFunction.myDimRBx

#### Widgets.SubFrameInterpFunction.myDimRBy

#### Widgets.SubFrameInterpFunction.myDimRBz

#### Widgets.SubFrameInterpFunction.myShapeC

#### Widgets.SubFrameInterpFunction.myShapeG

#### Widgets.SubFrameInterpFunction.myShapeL

#### Widgets.SubFrameInterpFunction.myShapeS

#### Widgets.SubFrameInterpFunction.preconEntry

#### Widgets.SubFrameInterpFunction.preconLabel

#### Widgets.SubFrameInterpFunction.shapeVal

#### The documentation for this class was generated from the following file:

#### Widgets.py

## Widgets.SubFrameXY Class Reference

Widget for XY label and entery.

Inheritance diagram for Widgets.SubFrameXY:

### Public Member Functions

**\_\_init\_\_** (self, container, parent, fieldNum, elementRow, elementCol, lableArr)

### Public Attributes

**cornerFramelongitude**

**latitude**

### Detailed Description

Widget for XY label and entery.

Longitude, Latitude have become interchangeable with X, Y

### Constructor & Destructor Documentation

#### Widgets.SubFrameXY.\_\_init\_\_ ( *self*, *container*, *parent*, *fieldNum*, *elementRow*, *elementCol*, *lableArr*)

### Member Data Documentation

#### Widgets.SubFrameXY.cornerFrame

#### Widgets.SubFrameXY.latitude

#### Widgets.SubFrameXY.longitude

#### The documentation for this class was generated from the following file:

#### Widgets.py

## shapefile.Writer Class Reference

Inheritance diagram for shapefile.Writer:

### Public Member Functions

**\_\_init\_\_** (self, **target**=None, **shapeType**=None, **autoBalance**=False, \*\*kwargs)

**\_\_len\_\_** (self)

**\_\_enter\_\_** (self)

**\_\_exit\_\_** (self, exc\_type, exc\_val, exc\_tb)

**\_\_del\_\_** (self)

**close** (self)

**shapeTypeName** (self)

**bbox** (self)

**zbox** (self)

**mbox** (self)

**shape** (self, s)

**record** (self, \*recordList, \*\*recordDict)

**balance** (self)

**null** (self)

**point** (self, x, y)

**pointm** (self, x, y, m=None)

**pointz** (self, x, y, z=0, m=None)

**multipoint** (self, points)

**multipointm** (self, points)

**multipointz** (self, points)

**line** (self, lines)

**linem** (self, lines)

**linez** (self, lines)

**poly** (self, polys)

**polym** (self, polys)

**polyz** (self, polys)

**multipatch** (self, parts, partTypes)

**field** (self, name, fieldType="C", size="50", decimal=0)

### Public Attributes

**targetautoBalance**

**fields**

**shapeType**

**shp**

**shx**

**dbf**

**recNum**

**shpNum**

**deletionFlag**

**encoding**

**encodingErrors**

### Protected Member Functions

**\_shapeparts** (self, parts, **shapeType**)

### Protected Attributes

**\_files\_to\_close\_bbox**

**\_zbox**

**\_mbox**

### Private Member Functions

**\_\_getFileObj** (self, f)

**\_\_shpFileLength** (self)

**\_\_bbox** (self, s)

**\_\_zbox** (self, s)

**\_\_mbox** (self, s)

**\_\_shapefileHeader** (self, fileObj, headerType='**shp**')

**\_\_dbfHeader** (self)

**\_\_shpRecord** (self, s)

**\_\_shxRecord** (self, offset, length)

**\_\_dbfRecord** (self, **record**)

### Detailed Description

Provides write support for ESRI Shapefiles.

### Constructor & Destructor Documentation

#### shapefile.Writer.\_\_init\_\_ ( *self*, *target* = None, *shapeType* = None, *autoBalance* = False, \*\* *kwargs*)

#### shapefile.Writer.\_\_del\_\_ ( *self*)

### Member Function Documentation

#### shapefile.Writer.\_\_bbox ( *self*, *s*)[private]

#### shapefile.Writer.\_\_dbfHeader ( *self*)[private]

Writes the dbf header and field descriptors.

#### shapefile.Writer.\_\_dbfRecord ( *self*, *record*)[private]

Writes the dbf records.

#### shapefile.Writer.\_\_enter\_\_ ( *self*)

Enter phase of context manager.

#### shapefile.Writer.\_\_exit\_\_ ( *self*, *exc\_type*, *exc\_val*, *exc\_tb*)

Exit phase of context manager, finish writing and close the files.

#### shapefile.Writer.\_\_getFileObj ( *self*, *f*)[private]

Safety handler to verify file-like objects

#### shapefile.Writer.\_\_len\_\_ ( *self*)

Returns the current number of features written to the shapefile.

If shapes and records are unbalanced, the length is considered the highest

of the two.

#### shapefile.Writer.\_\_mbox ( *self*, *s*)[private]

#### shapefile.Writer.\_\_shapefileHeader ( *self*, *fileObj*, *headerType* = 'shp')[private]

Writes the specified header type to the specified file-like object.

Several of the shapefile formats are so similar that a single generic

method to read or write them is warranted.

#### shapefile.Writer.\_\_shpFileLength ( *self*)[private]

Calculates the file length of the shp file.

#### shapefile.Writer.\_\_shpRecord ( *self*, *s*)[private]

#### shapefile.Writer.\_\_shxRecord ( *self*, *offset*, *length*)[private]

Writes the shx records.

#### shapefile.Writer.\_\_zbox ( *self*, *s*)[private]

#### shapefile.Writer.\_shapeparts ( *self*, *parts*, *shapeType*)[protected]

Internal method for adding a shape that has multiple collections of points (parts):

lines, polygons, and multipoint shapes.

#### shapefile.Writer.balance ( *self*)

Adds corresponding empty attributes or null geometry records depending

on which type of record was created to make sure all three files

are in synch.

#### shapefile.Writer.bbox ( *self*)

Returns the current bounding box for the shapefile which is

the lower-left and upper-right corners. It does not contain the

elevation or measure extremes.

#### shapefile.Writer.close ( *self*)

Write final shp, shx, and dbf headers, close opened files.

#### shapefile.Writer.field ( *self*, *name*, *fieldType* = "C", *size* = "50", *decimal* = 0)

Adds a dbf field descriptor to the shapefile.

#### shapefile.Writer.line ( *self*, *lines*)

Creates a POLYLINE shape.

Lines is a collection of lines, each made up of a list of xy values.

#### shapefile.Writer.linem ( *self*, *lines*)

Creates a POLYLINEM shape.

Lines is a collection of lines, each made up of a list of xym values.

If the m (measure) value is not included, it defaults to None (NoData).

#### shapefile.Writer.linez ( *self*, *lines*)

Creates a POLYLINEZ shape.

Lines is a collection of lines, each made up of a list of xyzm values.

If the z (elevation) value is not included, it defaults to 0.

If the m (measure) value is not included, it defaults to None (NoData).

#### shapefile.Writer.mbox ( *self*)

Returns the current m extremes for the shapefile.

#### shapefile.Writer.multipatch ( *self*, *parts*, *partTypes*)

Creates a MULTIPATCH shape.

Parts is a collection of 3D surface patches, each made up of a list of xyzm values.

PartTypes is a list of types that define each of the surface patches.

The types can be any of the following module constants: TRIANGLE\_STRIP,

TRIANGLE\_FAN, OUTER\_RING, INNER\_RING, FIRST\_RING, or RING.

If the z (elevation) value is not included, it defaults to 0.

If the m (measure) value is not included, it defaults to None (NoData).

#### shapefile.Writer.multipoint ( *self*, *points*)

Creates a MULTIPOINT shape.

Points is a list of xy values.

#### shapefile.Writer.multipointm ( *self*, *points*)

Creates a MULTIPOINTM shape.

Points is a list of xym values.

If the m (measure) value is not included, it defaults to None (NoData).

#### shapefile.Writer.multipointz ( *self*, *points*)

Creates a MULTIPOINTZ shape.

Points is a list of xyzm values.

If the z (elevation) value is not included, it defaults to 0.

If the m (measure) value is not included, it defaults to None (NoData).

#### shapefile.Writer.null ( *self*)

Creates a null shape.

#### shapefile.Writer.point ( *self*, *x*, *y*)

Creates a POINT shape.

#### shapefile.Writer.pointm ( *self*, *x*, *y*, *m* = None)

Creates a POINTM shape.

If the m (measure) value is not set, it defaults to NoData.

#### shapefile.Writer.pointz ( *self*, *x*, *y*, *z* = 0, *m* = None)

Creates a POINTZ shape.

If the z (elevation) value is not set, it defaults to 0.

If the m (measure) value is not set, it defaults to NoData.

#### shapefile.Writer.poly ( *self*, *polys*)

Creates a POLYGON shape.

Polys is a collection of polygons, each made up of a list of xy values.

Note that for ordinary polygons the coordinates must run in a clockwise direction.

If some of the polygons are holes, these must run in a counterclockwise direction.

#### shapefile.Writer.polym ( *self*, *polys*)

Creates a POLYGONM shape.

Polys is a collection of polygons, each made up of a list of xym values.

Note that for ordinary polygons the coordinates must run in a clockwise direction.

If some of the polygons are holes, these must run in a counterclockwise direction.

If the m (measure) value is not included, it defaults to None (NoData).

#### shapefile.Writer.polyz ( *self*, *polys*)

Creates a POLYGONZ shape.

Polys is a collection of polygons, each made up of a list of xyzm values.

Note that for ordinary polygons the coordinates must run in a clockwise direction.

If some of the polygons are holes, these must run in a counterclockwise direction.

If the z (elevation) value is not included, it defaults to 0.

If the m (measure) value is not included, it defaults to None (NoData).

#### shapefile.Writer.record ( *self*, \* *recordList*, \*\* *recordDict*)

Creates a dbf attribute record. You can submit either a sequence of

field values or keyword arguments of field names and values. Before

adding records you must add fields for the record values using the

field() method. If the record values exceed the number of fields the

extra ones won't be added. In the case of using keyword arguments to specify

field/value pairs only fields matching the already registered fields

will be added.

#### shapefile.Writer.shape ( *self*, *s*)

#### shapefile.Writer.shapeTypeName ( *self*)

#### shapefile.Writer.zbox ( *self*)

Returns the current z extremes for the shapefile.

### Member Data Documentation

#### shapefile.Writer.\_bbox[protected]

#### shapefile.Writer.\_files\_to\_close[protected]

#### shapefile.Writer.\_mbox[protected]

#### shapefile.Writer.\_zbox[protected]

#### shapefile.Writer.autoBalance

#### shapefile.Writer.dbf

#### shapefile.Writer.deletionFlag

#### shapefile.Writer.encoding

#### shapefile.Writer.encodingErrors

#### shapefile.Writer.fields

#### shapefile.Writer.recNum

#### shapefile.Writer.shapeType

#### shapefile.Writer.shp

#### shapefile.Writer.shpNum

#### shapefile.Writer.shx

#### shapefile.Writer.target

#### The documentation for this class was generated from the following file:

PyshpMaster/**shapefile.py**

# File Documentation

## AreaManager.py File Reference

### Classes

class **AreaManager.AreaManager***This class is used to paint area grouped by.*

class **AreaManager.Corner***Defines floating point data for corner defintions.*

### class AreaManager.AreaMgrSubFrameNamespaces

namespace **AreaManager**

## EditMathSetupFrame.py File Reference

### Classes

class **EditMathSetupFrame.EditMathSetup***This class allows the user to edit the Matlab/Octave setup files to fit their environment.*

### Namespaces

namespace **EditMathSetupFrame**

## FishMortBySpecAcc.py File Reference

### Classes

class **FishMortBySpecAcc.FishMortBySpecAcc***This class is used to assist the user in defining areas of interest to assess accumulated parameters located in these areas of interest.*

### Namespaces

namespace **FishMortBySpecAcc**

## GeoSams.py File Reference

### Classes

class **GeoSams.MainApplication***This class is the parent class for the GUI.*

### Namespaces

namespace **GeoSams**

### Functions

**GeoSams.ComputeResiduals** (obsFile, gridFile, procID, retDict)

**GeoSams.main** ()

## Globals.py File Reference

### Namespaces

namespace **Globals**

### Functions

**Globals.DetermineUnitsScale** (desiredParam)

**Globals.UpdateEntry** (entry, val)

**Globals.ShowMessage** (heading, message, type='info', timeout=2500)

*This method will display the message and then go away after the default time.*

### Variables

str **Globals.analDir** = 'Analysis'

str **Globals.configDir** = 'Configuration'

str **Globals.dataDir** = 'Data'

str **Globals.gridDir** = 'Grids'

str **Globals.interCfgDir** = 'Interpolation'

str **Globals.resultsDir** = 'Results'

str **Globals.shapeFileDir** = 'Shapefiles'

str **Globals.simCfgDir** = 'Simulation'

str **Globals.specAccCfgDir** = 'SpecialAccess'

str **Globals.surveyDataDir** = 'OriginalData'

list **Globals.comboTFStr** = ['T', 'F']

list **Globals.cornerLabelArr** = ['Corner', 'Long', 'Lat ', '0.0', '0.0']

int **Globals.frameWidth** = 400

int **Globals.frameHeight** = 200

int **Globals.scrollFrameHeight** = 600

int **Globals.helpXoffset** = 700

int **Globals.helpYoffset** = 50

int **Globals.meters\_per\_naut\_mile** = 1852

int **Globals.grid\_area\_sqm** = **meters\_per\_naut\_mile**\*\*2

str **Globals.ABUN** = 'ABUN\_'

str **Globals.BIOM** = 'BIOM\_'

str **Globals.EBMS** = 'EBMS\_'

str **Globals.FEFF** = 'FEFF\_'

str **Globals.FMOR** = 'FMOR\_'

str **Globals.LAND** = 'LAND\_'

str **Globals.LNDW** = 'LNDW\_'

str **Globals.LPUE** = 'LPUE\_'

str **Globals.RECR** = 'RECR\_'

int **Globals.scrollFrameWidth** = 900

str **Globals.geometryStr** = '920x725+10+10'

## GrowthFrame.py File Reference

### Classes

class **GrowthFrame.Growth***This class allows the user to adjust parameters used in computing scallop growth.*

### Namespaces

namespace **GrowthFrame**

## MainInputFrame.py File Reference

### Classes

class **MainInputFrame.MainInput***This class displays information about GeoSAMS simulation.*

### Namespaces

namespace **MainInputFrame**

## PointInPolygon.py File Reference

### Namespaces

namespace **PointInPolygon**

### Functions

**PointInPolygon.PointInPolygon** (polyX, polyY, x, y, nodes)

## PyshpMaster/shapefile.py File Reference

### Classes

class **shapefile.\_Array**class **shapefile.Shape**

class **shapefile.\_Record**

class **shapefile.ShapeRecord**

class **shapefile.Shapes**

class **shapefile.ShapeRecords**

class **shapefile.ShapefileException**

class **shapefile.Reader**

class **shapefile.Writer**

### Namespaces

namespace **shapefile**

### Functions

**shapefile.b** (v, encoding='utf-8', encodingErrors='strict')

**shapefile.u** (v, encoding='utf-8', encodingErrors='strict')

**shapefile.is\_string** (v)

**shapefile.pathlike\_obj** (path)

**shapefile.signed\_area** (coords, fast=False)

**shapefile.is\_cw** (coords)

**shapefile.rewind** (coords)

**shapefile.ring\_bbox** (coords)

**shapefile.bbox\_overlap** (bbox1, bbox2)

**shapefile.bbox\_contains** (bbox1, bbox2)

**shapefile.ring\_contains\_point** (coords, p)

**shapefile.ring\_sample** (coords, ccw=False)

**shapefile.ring\_contains\_ring** (coords1, coords2)

**shapefile.organize\_polygon\_rings** (rings, return\_errors=None)

**shapefile.test** (\*\*kwargs)

### Variables

str **shapefile.\_\_version\_\_** = "2.3.1"

**shapefile.logger** = logging.getLogger(\_\_name\_\_)

bool **shapefile.VERBOSE** = True

int **shapefile.NULL** = 0

int **shapefile.POINT** = 1

int **shapefile.POLYLINE** = 3

int **shapefile.POLYGON** = 5

int **shapefile.MULTIPOINT** = 8

int **shapefile.POINTZ** = 11

int **shapefile.POLYLINEZ** = 13

int **shapefile.POLYGONZ** = 15

int **shapefile.MULTIPOINTZ** = 18

int **shapefile.POINTM** = 21

int **shapefile.POLYLINEM** = 23

int **shapefile.POLYGONM** = 25

int **shapefile.MULTIPOINTM** = 28

int **shapefile.MULTIPATCH** = 31

dict **shapefile.SHAPETYPE\_LOOKUP**

int **shapefile.TRIANGLE\_STRIP** = 0

int **shapefile.TRIANGLE\_FAN** = 1

int **shapefile.OUTER\_RING** = 2

int **shapefile.INNER\_RING** = 3

int **shapefile.FIRST\_RING** = 4

int **shapefile.RING** = 5

dict **shapefile.PARTTYPE\_LOOKUP**

int **shapefile.PYTHON3** = 3

**shapefile.xrange** = range

**shapefile.izip** = zip

list **shapefile.MISSING** = [None,'']

int **shapefile.NODATA** = -10e38

**shapefile.failure\_count** = **test**()

## ShapeTest.py File Reference

### Classes

### class ShapeTest.GeoShapeNamespaces

namespace **ShapeTest**

### Variables

**ShapeTest.sf** = **shapefile.Reader**("Shapefiles/MAB\_Estimation\_Areas\_2019\_UTM18\_PDT.shp")

**ShapeTest.shapes** = sf.shapes()

**ShapeTest.shapeLen** = len(**sf**)

list **ShapeTest.shapeMA** = [ **GeoShape**() for \_ in range(**shapeLen**)]

**ShapeTest.record** = sf.record(n)

**ShapeTest.as\_dict** = record.as\_dict()

**ShapeTest.SAMS**

**ShapeTest.NewSAMS**

**ShapeTest.areaKm2**

**ShapeTest.pointLen** = len(**shapes**[n].points)

**ShapeTest.X**

**ShapeTest.Y**

**ShapeTest.lat**

**ShapeTest.lon**

list **ShapeTest.shapeGB** = [ **GeoShape**() for \_ in range(**shapeLen**)]

## SortByAreaFrame.py File Reference

### Classes

class **SortByAreaFrame.SortByArea***This class is used to assist the user in defining areas of interest to assess accumulated parameters located in these areas of interest.*

### Namespaces

namespace **SortByAreaFrame**

## SortByRegionFrame.py File Reference

### Classes

class **SortByRegionFrame.GeoShape***This class is used to define the shape of the regional data.*

class **SortByRegionFrame.SortByRegion***This class is used to assist the user in defining areas of interest to assess accumulated parameters located in these areas of interest.*

### Namespaces

namespace **SortByRegionFrame**

## SortIntoColumns.py File Reference

### Classes

class **SortIntoColumns.GeoShape**class **SortIntoColumns.Column**

### Namespaces

namespace **SortIntoColumns**

### Variables

**SortIntoColumns.inputFile** = sys.argv[1]

**SortIntoColumns.l** = len(**inputFile**)

**SortIntoColumns.domain** = **inputFile**[**l**-2:**l**]

**SortIntoColumns.dataFile** = os.path.join('Data', **inputFile**+'.csv')

**SortIntoColumns.outfile** = os.path.join('Data', **inputFile**+'\_BUFFER.csv')

**SortIntoColumns.M** = pd.read\_csv(**dataFile**)

**SortIntoColumns.fileName** = os.environ['GBShapeBufferFile']

str **SortIntoColumns.subDir** = 'GB\_Buffer'

**SortIntoColumns.shapeFile** = os.path.join('Shapefiles', **subDir**, **fileName**)

**SortIntoColumns.sf** = **shapefile.Reader**(**shapeFile**)

**SortIntoColumns.shapes** = sf.shapes()

**SortIntoColumns.shapeLen** = len(**sf**)

list **SortIntoColumns.shape** = [ **GeoShape**() for \_ in range(**shapeLen**)]

**SortIntoColumns.record** = sf.record(n)

**SortIntoColumns.as\_dict** = record.as\_dict()

**SortIntoColumns.Region**

**SortIntoColumns.pointLen** = len(**shapes**[n].points)

**SortIntoColumns.X**

**SortIntoColumns.Y**

list **SortIntoColumns.columns** = [**Column**() for \_ in range(**shapeLen**)]

**SortIntoColumns.name**

**SortIntoColumns.X\_t** = **M**['UTM\_X']

**SortIntoColumns.Y\_t** = **M**['UTM\_Y']

**SortIntoColumns.rows** = len(**X\_t**)

**SortIntoColumns.nodes** = len(**shape**[rgn].**X**)

**SortIntoColumns.sep**

**SortIntoColumns.na\_rep**

**SortIntoColumns.index**

## SpecialAreaFrame.py File Reference

### Classes

class **SpecialAreaFrame.SpecialArea***This class is used to assist the user in defining areas of interest to assess accumulated parameters located in these areas of interest.*

### Namespaces

namespace **SpecialAreaFrame**

## Widgets.py File Reference

### Classes

class **Widgets.SubFrameElement***Generic Element.*

class **Widgets.SubFrameInterpFunction**class **Widgets.SubFrameXY**

*Widget for XY label and entery.*

class **Widgets.ScrollFrame***Scrollable Frame Class from https://gist.github.com/mp035/9f2027c3ef9172264532fcd6262f3b01.*

### Namespaces

namespace **Widgets**

### Functions

**Widgets.numbersCallback** (input)

*Allows only correctly formed positive integers, ignores non-numeric characters.*

**Widgets.floatCallback** (input)

*Allows only correctly formed floats, ignores non-numeric characters.*

# Index

\_\_bbox

shapefile.Writer, 140

\_\_dbfHdrLength

shapefile.Reader, 101

\_\_dbfHeader

shapefile.Reader, 96

shapefile.Writer, 140

\_\_dbfRecord

shapefile.Writer, 140

\_\_del\_\_

shapefile.Reader, 96

shapefile.Writer, 140

\_\_dir\_\_

shapefile.\_Record, 46

\_\_enter\_\_

shapefile.Reader, 96

shapefile.Writer, 140

\_\_exit\_\_

shapefile.Reader, 97

shapefile.Writer, 141

\_\_field\_positions

shapefile.\_Record, 47

\_\_fieldLookup

shapefile.Reader, 101

\_\_fullRecLookup

shapefile.Reader, 101

\_\_fullRecStruct

shapefile.Reader, 101

\_\_geo\_interface\_\_

shapefile.Reader, 97

shapefile.Shape, 107

shapefile.ShapeRecord, 110

shapefile.ShapeRecords, 112

shapefile.Shapes, 113

\_\_getattr\_\_

shapefile.\_Record, 46

\_\_getFileObj

shapefile.Reader, 97

shapefile.Writer, 141

\_\_getitem\_\_

shapefile.\_Record, 46

\_\_init\_\_

AreaManager.AreaManager, 49

AreaManager.AreaMgrSubFrame, 52

AreaManager.Corner, 55

EditMathSetupFrame.EditMathSetup, 57

FishMortBySpecAcc.FishMortBySpecAcc, 60

GeoSams.MainApplication, 78

GrowthFrame.Growth, 71

MainInputFrame.MainInput, 88

shapefile.\_Record, 46

shapefile.Reader, 96

shapefile.Shape, 107

shapefile.ShapeRecord, 110

shapefile.Writer, 140

ShapeTest.GeoShape, 63

SortByAreaFrame.SortByArea, 117

SortByRegionFrame.GeoShape, 64

SortByRegionFrame.SortByRegion, 123

SortIntoColumns.Column, 54

SortIntoColumns.GeoShape, 65

SpecialAreaFrame.SpecialArea, 128

Widgets.ScrollFrame, 104

Widgets.SubFrameElement, 131

Widgets.SubFrameInterpFunction, 134

Widgets.SubFrameXY, 135

\_\_iter\_\_

shapefile.Reader, 97

\_\_len\_\_

shapefile.Reader, 97

shapefile.Writer, 141

\_\_mbox

shapefile.Writer, 141

\_\_oid

shapefile.\_Record, 47

shapefile.Shape, 108

\_\_record

shapefile.Reader, 97

\_\_recordFields

shapefile.Reader, 97

\_\_recordFmt

shapefile.Reader, 97

\_\_recordLength

shapefile.Reader, 101

\_\_repr\_\_

shapefile.\_Array, 44

shapefile.\_Record, 47

shapefile.Shape, 107

shapefile.ShapeRecords, 112

shapefile.Shapes, 113

\_\_restrictIndex

shapefile.Reader, 98

\_\_setattr\_\_

shapefile.\_Record, 47

\_\_setitem\_\_

shapefile.\_Record, 47

\_\_shape

shapefile.Reader, 98

\_\_shapefileHeader

shapefile.Writer, 141

\_\_shapeIndex

shapefile.Reader, 98

\_\_shpFileLength

shapefile.Writer, 141

\_\_shpHeader

shapefile.Reader, 98

\_\_shpRecord

shapefile.Writer, 141

\_\_shxHeader

shapefile.Reader, 98

\_\_shxOffsets

shapefile.Reader, 98

\_\_shxRecord

shapefile.Writer, 141

\_\_str\_\_

shapefile.Reader, 98

\_\_version\_\_

shapefile, 33

\_\_zbox

shapefile.Writer, 141

\_bbox

shapefile.Writer, 145

\_errors

shapefile.Shape, 108

\_files\_to\_close

shapefile.Reader, 101

shapefile.Writer, 145

\_from\_geojson

shapefile.Shape, 107

\_mbox

shapefile.Writer, 145

\_offsets

shapefile.Reader, 101

\_shapeparts

shapefile.Writer, 141

\_zbox

shapefile.Writer, 145

ABUN

Globals, 25

abunVar

MainInputFrame.MainInput, 90

addFrameClicked

GeoSams.MainApplication, 82

alphaMort

GrowthFrame.Growth, 72

alphStr

GrowthFrame.Growth, 72

analDir

Globals, 25

AppendResults

AreaManager.AreaMgrSubFrame, 52

AppendYears

SortByAreaFrame.SortByArea, 118

SortByRegionFrame.SortByRegion, 123

area

SortByRegionFrame.GeoShape, 64

areaData

AreaManager.AreaManager, 50

SortByAreaFrame.SortByArea, 119

areaFName

MainInputFrame.MainInput, 90

SortByAreaFrame.SortByArea, 119

SortByRegionFrame.SortByRegion, 125

SpecialAreaFrame.SpecialArea, 130

areaFrame

AreaManager.AreaMgrSubFrame, 53

areaKm2

ShapeTest, 36

ShapeTest.GeoShape, 63

SortByRegionFrame.SortByRegion, 125

AreaManager, 19

AreaManager.AreaManager, 48

\_\_init\_\_, 49

areaData, 50

areaSubFrame, 50

cornerRow, 50

numAreas, 50

numAreasMax, 50

NumAreasUpdate, 49

numCorners, 50

numCornersMax, 50

ReadAreaCorners, 49

scrollFrame, 50

sortAreaFrame, 50

UpdateWidgets, 49

AreaManager.AreaMgrSubFrame, 51

\_\_init\_\_, 52

AppendResults, 52

areaFrame, 53

comment, 53

commentEntry, 53

compAreaEntry, 53

corners, 53

EnterKeyClicked, 52

myArea, 53

numCorners, 53

numCornersEntry, 53

numCornersMax, 53

NumCornersUpdate, 52

numYearsMax, 53

numYrCols, 53

results, 53

SaveField, 52

startincCol, 53

yearStart, 53

yearStop, 53

AreaManager.Corner, 55

\_\_init\_\_, 55

lat, 55

long, 55

numCorners, 55

AreaManager.py, 146

areaMgr

FishMortBySpecAcc.FishMortBySpecAcc, 62

SpecialAreaFrame.SpecialArea, 130

areas

SortByAreaFrame.SortByArea, 119

areaSubFrame

AreaManager.AreaManager, 50

as\_dict

shapefile.\_Record, 47

ShapeTest, 36

SortIntoColumns, 40

autoBalance

shapefile.Writer, 145

b

shapefile, 31

balance

shapefile.Writer, 141

bbox

shapefile.Reader, 101

shapefile.Writer, 142

bbox\_contains

shapefile, 31

bbox\_overlap

shapefile, 31

BIOM

Globals, 25

bmsVar

MainInputFrame.MainInput, 90

browseExportButton

SortByAreaFrame.SortByArea, 119

SortByRegionFrame.SortByRegion, 125

BrowseExportFile

SortByAreaFrame.SortByArea, 118

SortByRegionFrame.SortByRegion, 123

canvas

Widgets.ScrollFrame, 105

canvas\_window

Widgets.ScrollFrame, 105

CBSelectedOutput

MainInputFrame.MainInput, 88

CbUpdateUnits

SortByAreaFrame.SortByArea, 118, 119

CheckStartDay

MainInputFrame.MainInput, 88, 90

CheckStopDay

MainInputFrame.MainInput, 88, 90

close

shapefile.Reader, 98

shapefile.Writer, 142

columns

SortIntoColumns, 40

comboParameter

SortByAreaFrame.SortByArea, 119

SortByRegionFrame.SortByRegion, 125

comboTFStr

Globals, 25

comment

AreaManager.AreaMgrSubFrame, 53

commentEntry

AreaManager.AreaMgrSubFrame, 53

compAreaEntry

AreaManager.AreaMgrSubFrame, 53

ComputeResiduals

GeoSams, 22

ComputeSelectOuputValue

MainInputFrame.MainInput, 88

configDir

Globals, 25

ConvertDayOfYrToMonthDay

GeoSams.MainApplication, 79

ConvertMonthDayToDayOfYr

GeoSams.MainApplication, 79

cornerFrame

Widgets.SubFrameXY, 136

cornerLabelArr

Globals, 25

cornerRow

AreaManager.AreaManager, 50

corners

AreaManager.AreaMgrSubFrame, 53

dataDir

Globals, 25

dataFile

SortIntoColumns, 40

dataSortFileEntry

SortByAreaFrame.SortByArea, 119

dataSortFileLabel

SortByAreaFrame.SortByArea, 119

dataSortUnitsEntry

SortByAreaFrame.SortByArea, 119

dataSortUnitsLabel

SortByAreaFrame.SortByArea, 119

daysInYear

GeoSams.MainApplication, 82

dbf

shapefile.Reader, 101

shapefile.Writer, 145

deletionFlag

shapefile.Writer, 145

desiredOutput

MainInputFrame.MainInput, 90

DetermineMgmtAreaIndex

SortByRegionFrame.SortByRegion, 123

DetermineUnitsScale

Globals, 23

dimVal

Widgets.SubFrameInterpFunction, 134

domain

SortIntoColumns, 40

domainName

GeoSams.MainApplication, 82

SortByAreaFrame.SortByArea, 119

SortByRegionFrame.SortByRegion, 125

domainNameCombo

MainInputFrame.MainInput, 90

domainNameLabel

MainInputFrame.MainInput, 90

dredgeDataFile

MainInputFrame.MainInput, 90

dredgeWdStr

GrowthFrame.Growth, 72

dredgeWth

GrowthFrame.Growth, 72

EBMS

Globals, 25

ebmsVar

MainInputFrame.MainInput, 90

editMathFrame

EditMathSetupFrame.EditMathSetup, 58

EditMathSetupFrame, 20

EditMathSetupFrame.EditMathSetup, 56

\_\_init\_\_, 57

editMathFrame, 58

editText, 58

editTextButton, 58

isMatlab, 58

LoadStartupFile, 57

matlabFName, 58

octaveFName, 58

on\_visibility, 57, 58

pop\_up, 57

ReadFile, 57

useMatlabRB, 58

useOctaveRB, 58

usingMatlab, 58

WriteStartupFile, 57

EditMathSetupFrame.py, 147

editText

EditMathSetupFrame.EditMathSetup, 58

editTextButton

EditMathSetupFrame.EditMathSetup, 58

encoding

shapefile.Reader, 101

shapefile.Writer, 145

encodingErrors

shapefile.Reader, 101

shapefile.Writer, 145

EnterKeyClicked

AreaManager.AreaMgrSubFrame, 52

FishMortBySpecAcc.FishMortBySpecAcc, 61, 62

MainInputFrame.MainInput, 88

SortByAreaFrame.SortByArea, 118, 119

SpecialAreaFrame.SpecialArea, 129, 130

ExportAll

SortByAreaFrame.SortByArea, 118

SortByRegionFrame.SortByRegion, 123

exportAllSortButton

SortByAreaFrame.SortByArea, 119

SortByRegionFrame.SortByRegion, 125

exportFileEntry

SortByAreaFrame.SortByArea, 119

SortByRegionFrame.SortByRegion, 125

exportFileLabel

SortByAreaFrame.SortByArea, 119

SortByRegionFrame.SortByRegion, 125

exportFileName

SortByAreaFrame.SortByArea, 119

SortByRegionFrame.SortByRegion, 125

ExportThis

SortByAreaFrame.SortByArea, 118

SortByRegionFrame.SortByRegion, 124

exportThisSortButton

SortByAreaFrame.SortByArea, 119

SortByRegionFrame.SortByRegion, 125

failure\_count

shapefile, 33

FEFF

Globals, 25

feffVar

MainInputFrame.MainInput, 90

field

shapefile.Writer, 142

fields

shapefile.Reader, 101

shapefile.Writer, 145

fileName

SortIntoColumns, 40

FIRST\_RING

shapefile, 33

firstYrCol

SortByRegionFrame.SortByRegion, 125

fishMort

GrowthFrame.Growth, 72

FishMortBySpecAcc, 21

FishMortBySpecAcc.FishMortBySpecAcc, 59

\_\_init\_\_, 60

areaMgr, 62

EnterKeyClicked, 61, 62

fishMortFile, 62

fishMortFileLabel, 62

fmFName, 62

GetFMFile, 61

numAreas, 62

numDefined, 62

numDefinedEntry, 62

numDefinedLabel, 62

numDefinedMax, 62

NumDefinedUpdate, 61

numFieldsMax, 62

numFieldss, 62

on\_visibility, 61, 62

openFMFileButton, 62

pop\_up, 61

ReadFields, 61

root, 62

SaveFMFile, 61

saveFMFileButton, 62

startDir, 62

UpdateWidgets, 61

yearEntry, 62

FishMortBySpecAcc.py, 148

fishMortFile

FishMortBySpecAcc.FishMortBySpecAcc, 62

fishMortFileLabel

FishMortBySpecAcc.FishMortBySpecAcc, 62

floatCallback

Widgets, 43

fmFName

FishMortBySpecAcc.FishMortBySpecAcc, 62

FMOR

Globals, 25

fmortFileName

GrowthFrame.Growth, 72

fmortStr

GrowthFrame.Growth, 72

fmortVar

MainInputFrame.MainInput, 90

frame1

GeoSams.MainApplication, 82

frame2

GeoSams.MainApplication, 82

frame3

GeoSams.MainApplication, 82

frame5

GeoSams.MainApplication, 82

frame6

GeoSams.MainApplication, 82

frame7

GeoSams.MainApplication, 82

frame8

GeoSams.MainApplication, 82

frameHeight

Globals, 25

frameWidth

Globals, 25

friend

GrowthFrame.Growth, 72

MainInputFrame.MainInput, 90

SortByAreaFrame.SortByArea, 119

SortByRegionFrame.SortByRegion, 125

SpecialAreaFrame.SpecialArea, 130

funcFrame

Widgets.SubFrameInterpFunction, 134

gbAdultMort

GrowthFrame.Growth, 72

gbAdultMortStr

GrowthFrame.Growth, 72

gbClFSelAStr

GrowthFrame.Growth, 72

gbClFSelBStr

GrowthFrame.Growth, 72

gbClosedFSelectA

GrowthFrame.Growth, 72

gbClosedFSelectB

GrowthFrame.Growth, 72

gbCullSize

GrowthFrame.Growth, 72

gbCullStr

GrowthFrame.Growth, 72

gbDiscard

GrowthFrame.Growth, 72

gbDiscStr

GrowthFrame.Growth, 72

gbIncident

GrowthFrame.Growth, 72

gbIncidStr

GrowthFrame.Growth, 72

gbLen0Str

GrowthFrame.Growth, 72

gbLength0

GrowthFrame.Growth, 72

gbOpenFSelectA

GrowthFrame.Growth, 72

gbOpenFSelectB

GrowthFrame.Growth, 73

gbOpFSelAStr

GrowthFrame.Growth, 73

gbOpFSelBStr

GrowthFrame.Growth, 73

gbShapeBufferFileEntry

MainInputFrame.MainInput, 90

gbShapeFileEntry

MainInputFrame.MainInput, 90

geometryStr

Globals, 25

GeoSams, 22

ComputeResiduals, 22

main, 22

GeoSAMS GUI, 2

GeoSams.MainApplication, 75

\_\_init\_\_, 78

addFrameClicked, 82

ConvertDayOfYrToMonthDay, 79

ConvertMonthDayToDayOfYr, 79

daysInYear, 82

domainName, 82

frame1, 82

frame2, 82

frame3, 82

frame5, 82

frame6, 82

frame7, 82

frame8, 82

gmConfigFile, 82

InterpAndPlotResults, 79

isSkip, 82

maxAreas, 82

maxCorners, 82

maxYears, 82

monDict, 82

notebook, 82

paramStr, 82

paramVal, 82

pop\_up, 79

ReadConfigFile, 79

ReadGridMgrConfigFile, 80

ReadRecruitConfigFile, 80

ReadSimConfigFile, 80

recrConfigFile, 82

recruitYrStop, 82

recruitYrStrt, 82

root, 82

Run\_Sim, 80

SaveConfigFiles, 80

ShowArgs, 80

simConfigFile, 83

skipStatusMsgs, 83

skipStatusMsgsRB, 83

specAccFileStr, 83

style, 83

ToggleSkipStatusMsgs, 80

tsPerYear, 83

useHabCamData, 83

WriteGridMgrConfig, 80

WriteGrowthConfig, 80

WriteRecruitmentConfig, 80

WriteScallopConfig, 80

WriteSpatialFncsConfig, 81

yearStart, 83

yearStop, 83

GeoSams.py, 149

GetAreaFile

SpecialAreaFrame.SpecialArea, 129

GetDataSortFile

SortByAreaFrame.SortByArea, 118

GetFMFile

FishMortBySpecAcc.FishMortBySpecAcc, 61

GetGMgrConfigFName

SpecialAreaFrame.SpecialArea, 129

GetGrowthConfigFName

GrowthFrame.Growth, 71

GetRecrConfigFName

MainInputFrame.MainInput, 88

GetSelectedOutputs

MainInputFrame.MainInput, 88

GetSimConfigFName

MainInputFrame.MainInput, 88

GetZoneData

SortByRegionFrame.SortByRegion, 124

Globals, 23

ABUN, 25

analDir, 25

BIOM, 25

comboTFStr, 25

configDir, 25

cornerLabelArr, 25

dataDir, 25

DetermineUnitsScale, 23

EBMS, 25

FEFF, 25

FMOR, 25

frameHeight, 25

frameWidth, 25

geometryStr, 25

grid\_area\_sqm, 25

gridDir, 25

helpXoffset, 25

helpYoffset, 25

interCfgDir, 25

LAND, 25

LNDW, 25

LPUE, 25

meters\_per\_naut\_mile, 25

RECR, 25

resultsDir, 26

scrollFrameHeight, 26

scrollFrameWidth, 26

shapeFileDir, 26

ShowMessage, 23

simCfgDir, 26

specAccCfgDir, 26

surveyDataDir, 26

UpdateEntry, 24

Globals.py, 150

gmCfgFile

SpecialAreaFrame.SpecialArea, 130

gmConfigFile

GeoSams.MainApplication, 82

grid\_area\_sqm

Globals, 25

gridDir

Globals, 25

Growth Frame, 6

growthCfgFile

GrowthFrame.Growth, 73

GrowthFrame, 27

GrowthFrame.Growth, 66

\_\_init\_\_, 71

alphaMort, 72

alphStr, 72

dredgeWdStr, 72

dredgeWth, 72

fishMort, 72

fmortFileName, 72

fmortStr, 72

friend, 72

gbAdultMort, 72

gbAdultMortStr, 72

gbClFSelAStr, 72

gbClFSelBStr, 72

gbClosedFSelectA, 72

gbClosedFSelectB, 72

gbCullSize, 72

gbCullStr, 72

gbDiscard, 72

gbDiscStr, 72

gbIncident, 72

gbIncidStr, 72

gbLen0Str, 72

gbLength0, 72

gbOpenFSelectA, 72

gbOpenFSelectB, 73

gbOpFSelAStr, 73

gbOpFSelBStr, 73

GetGrowthConfigFName, 71

growthCfgFile, 73

growthStartDir, 73

loadGrowthConfigButton, 73

LoadGrowthData, 71

lpueIntcept, 73

lpueIntcStr, 73

lpueSl2Str, 73

lpueSlope, 73

lpueSlope2, 73

lpueSlStr, 73

maAdultMort, 73

maAdultMortStr, 73

maCullSize, 73

maCullStr, 73

maDiscard, 73

maDiscStr, 73

maFSelAStr, 73

maFSelBStr, 73

maFSelectA, 73

maFSelectB, 73

maIncident, 73

maIncidStr, 73

maLen0Str, 73

maLength0, 74

maxPerDay, 74

maxPerDayStr, 74

maxTime, 74

maxTimeStr, 74

on\_visibility, 71, 74

pop\_up, 71

root, 74

saveGrowthConfigButton, 74

towSpdStr, 74

towSpeed, 74

UpdateValues, 71

UpdateWidgets, 71

GrowthFrame.py, 151

growthStartDir

GrowthFrame.Growth, 73

habCamDataFile

MainInputFrame.MainInput, 90

helpXoffset

Globals, 25

helpYoffset

Globals, 25

inBox

SortIntoColumns.Column, 54

index

SortIntoColumns, 40

INNER\_RING

shapefile, 33

inputFile

SortIntoColumns, 40

interCfgDir

Globals, 25

InterpAndPlotResults

GeoSams.MainApplication, 79

interpStartDir

MainInputFrame.MainInput, 90

is\_cw

shapefile, 31

is\_string

shapefile, 31

isMatlab

EditMathSetupFrame.EditMathSetup, 58

isSkip

GeoSams.MainApplication, 82

iterRecords

shapefile.Reader, 98

iterShapeRecords

shapefile.Reader, 98

iterShapes

shapefile.Reader, 99

izip

shapefile, 33

l

SortIntoColumns, 40

LAND

Globals, 25

landVar

MainInputFrame.MainInput, 90

lat

AreaManager.Corner, 55

ShapeTest, 36

ShapeTest.GeoShape, 63

SortByRegionFrame.GeoShape, 64

SortIntoColumns.GeoShape, 65

latitude

Widgets.SubFrameXY, 136

line

shapefile.Writer, 142

linem

shapefile.Writer, 142

linez

shapefile.Writer, 142

LNDW

Globals, 25

lndwVar

MainInputFrame.MainInput, 90

load

shapefile.Reader, 99

load\_dbf

shapefile.Reader, 99

load\_shp

shapefile.Reader, 99

load\_shx

shapefile.Reader, 99

loadGrowthConfigButton

GrowthFrame.Growth, 73

LoadGrowthData

GrowthFrame.Growth, 71

LoadStartupFile

EditMathSetupFrame.EditMathSetup, 57

logger

shapefile, 33

lon

ShapeTest, 36

ShapeTest.GeoShape, 63

SortByRegionFrame.GeoShape, 64

SortIntoColumns.GeoShape, 65

long

AreaManager.Corner, 55

longitude

Widgets.SubFrameXY, 136

LPUE

Globals, 25

lpueIntcept

GrowthFrame.Growth, 73

lpueIntcStr

GrowthFrame.Growth, 73

lpueSl2Str

GrowthFrame.Growth, 73

lpueSlope

GrowthFrame.Growth, 73

lpueSlope2

GrowthFrame.Growth, 73

lpueSlStr

GrowthFrame.Growth, 73

lpueVar

MainInputFrame.MainInput, 90

M

SortIntoColumns, 40

maAdultMort

GrowthFrame.Growth, 73

maAdultMortStr

GrowthFrame.Growth, 73

maCullSize

GrowthFrame.Growth, 73

maCullStr

GrowthFrame.Growth, 73

maDiscard

GrowthFrame.Growth, 73

maDiscStr

GrowthFrame.Growth, 73

maFSelAStr

GrowthFrame.Growth, 73

maFSelBStr

GrowthFrame.Growth, 73

maFSelectA

GrowthFrame.Growth, 73

maFSelectB

GrowthFrame.Growth, 73

main

GeoSams, 22

Main, 9

maIncident

GrowthFrame.Growth, 73

maIncidStr

GrowthFrame.Growth, 73

MainInputFrame, 28

MainInputFrame.MainInput, 84

\_\_init\_\_, 88

abunVar, 90

areaFName, 90

bmsVar, 90

CBSelectedOutput, 88

CheckStartDay, 88, 90

CheckStopDay, 88, 90

ComputeSelectOuputValue, 88

desiredOutput, 90

domainNameCombo, 90

domainNameLabel, 90

dredgeDataFile, 90

ebmsVar, 90

EnterKeyClicked, 88

feffVar, 90

fmortVar, 90

friend, 90

gbShapeBufferFileEntry, 90

gbShapeFileEntry, 90

GetRecrConfigFName, 88

GetSelectedOutputs, 88

GetSimConfigFName, 88

habCamDataFile, 90

interpStartDir, 90

landVar, 90

lndwVar, 90

lpueVar, 90

maShapeBufferFileEntry, 90

maShapeFileEntry, 90

maxYears, 90

monthsArr, 91

numYrsAvg, 91

OpenPDF, 89

openPDFButton, 91

openRecrConfigButton, 91

openSimConfigButton, 91

pop\_up, 89

recrCfgFile, 91

recrVar, 91

recrYrStop, 91

recrYrStrt, 91

root, 91

setDredgeDataButton, 91

SetDredgeFileEnvVar, 89

SetDredgeFileName, 89

SetGbShapeBufferFile, 89

setGbShapeBufferFileButton, 91

SetGBShapeBufferFileEnvVar, 89

SetGbShapeFile, 89

setGbShapeFileButton, 91

SetGBShapeFileEnvVar, 89

setHabCamDataButton, 91

SetHabCamFileEnvVar, 89

SetHabCamFileName, 89

SetMaShapeBufferFile, 89

setMaShapeBufferFileButton, 91

SetMaShapeBufferFileEnvVar, 89

SetMaShapeFile, 89

setMaShapeFileButton, 91

SetMaShapeFileEnvVar, 89

shapeFileDir, 91

simCfgFile, 91

simStartDir, 91

startDayComboDay, 91

startDayComboMonth, 91

startDayLabel, 91

startYr, 91

stopDayComboDay, 91

stopDayComboMonth, 91

stopDayLabel, 92

stopYr, 92

surveyStartDir, 92

tsPerYear, 92

MainInputFrame.py, 152

maLen0Str

GrowthFrame.Growth, 73

maLength0

GrowthFrame.Growth, 74

maShapeBufferFileEntry

MainInputFrame.MainInput, 90

maShapeFileEntry

MainInputFrame.MainInput, 90

Math Setup Frame, 4

matlabFName

EditMathSetupFrame.EditMathSetup, 58

maxAreas

GeoSams.MainApplication, 82

SortByRegionFrame.SortByRegion, 125

maxCorners

GeoSams.MainApplication, 82

maxPerDay

GrowthFrame.Growth, 74

maxPerDayStr

GrowthFrame.Growth, 74

maxTime

GrowthFrame.Growth, 74

maxTimeStr

GrowthFrame.Growth, 74

maxYears

GeoSams.MainApplication, 82

MainInputFrame.MainInput, 90

SortByAreaFrame.SortByArea, 119

SortByRegionFrame.SortByRegion, 125

mbox

shapefile.Reader, 101

shapefile.Writer, 142

meters\_per\_naut\_mile

Globals, 25

MISSING

shapefile, 33

monDict

GeoSams.MainApplication, 82

monthsArr

MainInputFrame.MainInput, 91

multipatch

shapefile.Writer, 142

MULTIPATCH

shapefile, 33

multipoint

shapefile.Writer, 143

MULTIPOINT

shapefile, 33

multipointm

shapefile.Writer, 143

MULTIPOINTM

shapefile, 33

multipointz

shapefile.Writer, 143

MULTIPOINTZ

shapefile, 33

myArea

AreaManager.AreaMgrSubFrame, 53

myDimRBx

Widgets.SubFrameInterpFunction, 134

myDimRBy

Widgets.SubFrameInterpFunction, 134

myDimRBz

Widgets.SubFrameInterpFunction, 134

myEntry

Widgets.SubFrameElement, 132

myLabel

Widgets.SubFrameElement, 132

myShapeC

Widgets.SubFrameInterpFunction, 134

myShapeG

Widgets.SubFrameInterpFunction, 134

myShapeL

Widgets.SubFrameInterpFunction, 134

myShapeS

Widgets.SubFrameInterpFunction, 134

na\_rep

SortIntoColumns, 40

name

SortIntoColumns, 40

SortIntoColumns.Column, 54

NewSAMS

ShapeTest, 36

ShapeTest.GeoShape, 63

NODATA

shapefile, 33

nodes

SortIntoColumns, 40

notebook

GeoSams.MainApplication, 82

null

shapefile.Writer, 143

NULL

shapefile, 33

numAreas

AreaManager.AreaManager, 50

FishMortBySpecAcc.FishMortBySpecAcc, 62

SortByAreaFrame.SortByArea, 119

SortByRegionFrame.SortByRegion, 125

SpecialAreaFrame.SpecialArea, 130

numAreasEntry

SortByAreaFrame.SortByArea, 120

SpecialAreaFrame.SpecialArea, 130

numAreasLabel

SortByAreaFrame.SortByArea, 120

SpecialAreaFrame.SpecialArea, 130

numAreasMax

AreaManager.AreaManager, 50

SortByAreaFrame.SortByArea, 120

SpecialAreaFrame.SpecialArea, 130

NumAreasUpdate

AreaManager.AreaManager, 49

SortByAreaFrame.SortByArea, 118

SpecialAreaFrame.SpecialArea, 129

numbersCallback

Widgets, 43

numCorners

AreaManager.AreaManager, 50

AreaManager.AreaMgrSubFrame, 53

AreaManager.Corner, 55

SortByAreaFrame.SortByArea, 120

SortByRegionFrame.SortByRegion, 125

SpecialAreaFrame.SpecialArea, 130

numCornersEntry

AreaManager.AreaMgrSubFrame, 53

numCornersMax

AreaManager.AreaManager, 50

AreaManager.AreaMgrSubFrame, 53

SortByAreaFrame.SortByArea, 120

SpecialAreaFrame.SpecialArea, 130

NumCornersUpdate

AreaManager.AreaMgrSubFrame, 52

numDefined

FishMortBySpecAcc.FishMortBySpecAcc, 62

numDefinedEntry

FishMortBySpecAcc.FishMortBySpecAcc, 62

numDefinedLabel

FishMortBySpecAcc.FishMortBySpecAcc, 62

numDefinedMax

FishMortBySpecAcc.FishMortBySpecAcc, 62

NumDefinedUpdate

FishMortBySpecAcc.FishMortBySpecAcc, 61

numFieldsMax

FishMortBySpecAcc.FishMortBySpecAcc, 62

numFieldss

FishMortBySpecAcc.FishMortBySpecAcc, 62

numRecords

shapefile.Reader, 101

numShapes

shapefile.Reader, 101

numYears

SortByAreaFrame.SortByArea, 120

SortByRegionFrame.SortByRegion, 125

numYearsMax

AreaManager.AreaMgrSubFrame, 53

numYrCols

AreaManager.AreaMgrSubFrame, 53

numYrsAvg

MainInputFrame.MainInput, 91

octaveFName

EditMathSetupFrame.EditMathSetup, 58

oid

shapefile.\_Record, 47

shapefile.Shape, 107

on\_visibility

EditMathSetupFrame.EditMathSetup, 57, 58

FishMortBySpecAcc.FishMortBySpecAcc, 61, 62

GrowthFrame.Growth, 71, 74

SortByAreaFrame.SortByArea, 118, 120

SortByRegionFrame.SortByRegion, 124, 125

SpecialAreaFrame.SpecialArea, 129, 130

onEnter

Widgets.ScrollFrame, 104, 105

onFrameConfigure

Widgets.ScrollFrame, 104, 105

onLeave

Widgets.ScrollFrame, 104, 105

onMouseWheel

Widgets.ScrollFrame, 104, 105

openAreaFileButton

SpecialAreaFrame.SpecialArea, 130

openDataSortButton

SortByAreaFrame.SortByArea, 120

openFMFileButton

FishMortBySpecAcc.FishMortBySpecAcc, 62

openGmgrConfigButton

SpecialAreaFrame.SpecialArea, 130

OpenPDF

MainInputFrame.MainInput, 89

openPDFButton

MainInputFrame.MainInput, 91

openRecrConfigButton

MainInputFrame.MainInput, 91

openSimConfigButton

MainInputFrame.MainInput, 91

organize\_polygon\_rings

shapefile, 31

OUTER\_RING

shapefile, 33

outfile

SortIntoColumns, 40

outputParmLabel

SortByAreaFrame.SortByArea, 120

SortByRegionFrame.SortByRegion, 125

paramStr

GeoSams.MainApplication, 82

SortByAreaFrame.SortByArea, 120

SortByRegionFrame.SortByRegion, 125

paramVal

GeoSams.MainApplication, 82

parts

shapefile.Shape, 108

PARTTYPE\_LOOKUP

shapefile, 33

partTypes

shapefile.Shape, 108

pathlike\_obj

shapefile, 32

point

shapefile.Writer, 143

POINT

shapefile, 34

PointInPolygon, 29

PointInPolygon, 29

PointInPolygon.py, 153

pointLen

ShapeTest, 36

SortIntoColumns, 40

pointm

shapefile.Writer, 143

POINTM

shapefile, 34

points

shapefile.Shape, 108

pointz

shapefile.Writer, 143

POINTZ

shapefile, 34

poly

shapefile.Writer, 143

POLYGON

shapefile, 34

POLYGONM

shapefile, 34

POLYGONZ

shapefile, 34

POLYLINE

shapefile, 34

POLYLINEM

shapefile, 34

POLYLINEZ

shapefile, 34

polym

shapefile.Writer, 144

polyz

shapefile.Writer, 144

pop\_up

EditMathSetupFrame.EditMathSetup, 57

FishMortBySpecAcc.FishMortBySpecAcc, 61

GeoSams.MainApplication, 79

GrowthFrame.Growth, 71

MainInputFrame.MainInput, 89

SortByAreaFrame.SortByArea, 118

SortByRegionFrame.SortByRegion, 124

SpecialAreaFrame.SpecialArea, 129

preconEntry

Widgets.SubFrameInterpFunction, 134

preconLabel

Widgets.SubFrameInterpFunction, 134

PyshpMaster/shapefile.py, 154

PYTHON3

shapefile, 34

ReadAreaCorners

AreaManager.AreaManager, 49

ReadConfigFile

GeoSams.MainApplication, 79

ReadFields

FishMortBySpecAcc.FishMortBySpecAcc, 61

ReadFile

EditMathSetupFrame.EditMathSetup, 57

ReadGridMgrConfigFile

GeoSams.MainApplication, 80

ReadRecruitConfigFile

GeoSams.MainApplication, 80

ReadSimConfigFile

GeoSams.MainApplication, 80

recNum

shapefile.Writer, 145

record

shapefile.Reader, 99

shapefile.ShapeRecord, 111

shapefile.Writer, 144

ShapeTest, 36

SortIntoColumns, 40

records

shapefile.Reader, 99

RECR

Globals, 25

recrCfgFile

MainInputFrame.MainInput, 91

recrConfigFile

GeoSams.MainApplication, 82

recruitYrStop

GeoSams.MainApplication, 82

recruitYrStrt

GeoSams.MainApplication, 82

recrVar

MainInputFrame.MainInput, 91

recrYrStop

MainInputFrame.MainInput, 91

recrYrStrt

MainInputFrame.MainInput, 91

Region

SortIntoColumns, 40

SortIntoColumns.GeoShape, 65

results

AreaManager.AreaMgrSubFrame, 53

resultsDir

Globals, 26

SortByRegionFrame.SortByRegion, 125

rewind

shapefile, 32

RING

shapefile, 34

ring\_bbox

shapefile, 32

ring\_contains\_point

shapefile, 32

ring\_contains\_ring

shapefile, 32

ring\_sample

shapefile, 32

root

FishMortBySpecAcc.FishMortBySpecAcc, 62

GeoSams.MainApplication, 82

GrowthFrame.Growth, 74

MainInputFrame.MainInput, 91

SortByAreaFrame.SortByArea, 120

SortByRegionFrame.SortByRegion, 125

SpecialAreaFrame.SpecialArea, 130

rows

SortIntoColumns, 40

Run\_Sim

GeoSams.MainApplication, 80

RunSort

SortByAreaFrame.SortByArea, 118

SortByRegionFrame.SortByRegion, 124

runSortButton

SortByAreaFrame.SortByArea, 120

SortByRegionFrame.SortByRegion, 125

SAMS

ShapeTest, 36

ShapeTest.GeoShape, 63

SaveAreaFile

SpecialAreaFrame.SpecialArea, 129

saveAreaFileButton

SpecialAreaFrame.SpecialArea, 130

SaveConfigFiles

GeoSams.MainApplication, 80

saveDataSortButton

SortByAreaFrame.SortByArea, 120

SaveDataSortFile

SortByAreaFrame.SortByArea, 118

SaveField

AreaManager.AreaMgrSubFrame, 52

SaveFMFile

FishMortBySpecAcc.FishMortBySpecAcc, 61

saveFMFileButton

FishMortBySpecAcc.FishMortBySpecAcc, 62

saveGrowthConfigButton

GrowthFrame.Growth, 74

SaveSpecialAreaData

SpecialAreaFrame.SpecialArea, 129

scrollFrame

AreaManager.AreaManager, 50

SortByAreaFrame.SortByArea, 120

SortByRegionFrame.SortByRegion, 126

scrollFrameHeight

Globals, 26

scrollFrameWidth

Globals, 26

sep

SortIntoColumns, 40

Set Fishing Mortality in Special Access Areas, 5

setDredgeDataButton

MainInputFrame.MainInput, 91

SetDredgeFileEnvVar

MainInputFrame.MainInput, 89

SetDredgeFileName

MainInputFrame.MainInput, 89

SetGbShapeBufferFile

MainInputFrame.MainInput, 89

setGbShapeBufferFileButton

MainInputFrame.MainInput, 91

SetGBShapeBufferFileEnvVar

MainInputFrame.MainInput, 89

SetGbShapeFile

MainInputFrame.MainInput, 89

setGbShapeFileButton

MainInputFrame.MainInput, 91

SetGBShapeFileEnvVar

MainInputFrame.MainInput, 89

setHabCamDataButton

MainInputFrame.MainInput, 91

SetHabCamFileEnvVar

MainInputFrame.MainInput, 89

SetHabCamFileName

MainInputFrame.MainInput, 89

SetMaShapeBufferFile

MainInputFrame.MainInput, 89

setMaShapeBufferFileButton

MainInputFrame.MainInput, 91

SetMaShapeBufferFileEnvVar

MainInputFrame.MainInput, 89

SetMaShapeFile

MainInputFrame.MainInput, 89

setMaShapeFileButton

MainInputFrame.MainInput, 91

SetMaShapeFileEnvVar

MainInputFrame.MainInput, 89

sf

ShapeTest, 36

SortIntoColumns, 40

shape

shapefile.Reader, 100

shapefile.ShapeRecord, 111

shapefile.Writer, 144

SortIntoColumns, 40

shapefile, 30

\_\_version\_\_, 33

b, 31

bbox\_contains, 31

bbox\_overlap, 31

failure\_count, 33

FIRST\_RING, 33

INNER\_RING, 33

is\_cw, 31

is\_string, 31

izip, 33

logger, 33

MISSING, 33

MULTIPATCH, 33

MULTIPOINT, 33

MULTIPOINTM, 33

MULTIPOINTZ, 33

NODATA, 33

NULL, 33

organize\_polygon\_rings, 31

OUTER\_RING, 33

PARTTYPE\_LOOKUP, 33

pathlike\_obj, 32

POINT, 34

POINTM, 34

POINTZ, 34

POLYGON, 34

POLYGONM, 34

POLYGONZ, 34

POLYLINE, 34

POLYLINEM, 34

POLYLINEZ, 34

PYTHON3, 34

rewind, 32

RING, 34

ring\_bbox, 32

ring\_contains\_point, 32

ring\_contains\_ring, 32

ring\_sample, 32

SHAPETYPE\_LOOKUP, 34

signed\_area, 32

test, 33

TRIANGLE\_FAN, 34

TRIANGLE\_STRIP, 34

u, 33

VERBOSE, 34

xrange, 34

shapeFile

SortIntoColumns, 40

shapefile.\_Array, 44

\_\_repr\_\_, 44

shapefile.\_Record, 45

\_\_dir\_\_, 46

\_\_field\_positions, 47

\_\_getattr\_\_, 46

\_\_getitem\_\_, 46

\_\_init\_\_, 46

\_\_oid, 47

\_\_repr\_\_, 47

\_\_setattr\_\_, 47

\_\_setitem\_\_, 47

as\_dict, 47

oid, 47

shapefile.Reader, 93

\_\_dbfHdrLength, 101

\_\_dbfHeader, 96

\_\_del\_\_, 96

\_\_enter\_\_, 96

\_\_exit\_\_, 97

\_\_fieldLookup, 101

\_\_fullRecLookup, 101

\_\_fullRecStruct, 101

\_\_geo\_interface\_\_, 97

\_\_getFileObj, 97

\_\_init\_\_, 96

\_\_iter\_\_, 97

\_\_len\_\_, 97

\_\_record, 97

\_\_recordFields, 97

\_\_recordFmt, 97

\_\_recordLength, 101

\_\_restrictIndex, 98

\_\_shape, 98

\_\_shapeIndex, 98

\_\_shpHeader, 98

\_\_shxHeader, 98

\_\_shxOffsets, 98

\_\_str\_\_, 98

\_files\_to\_close, 101

\_offsets, 101

bbox, 101

close, 98

dbf, 101

encoding, 101

encodingErrors, 101

fields, 101

iterRecords, 98

iterShapeRecords, 98

iterShapes, 99

load, 99

load\_dbf, 99

load\_shp, 99

load\_shx, 99

mbox, 101

numRecords, 101

numShapes, 101

record, 99

records, 99

shape, 100

shapeName, 101

shapeRecord, 100

shapeRecords, 100

shapes, 100

shapeType, 101

shapeTypeName, 100

shp, 101

shpLength, 101

shx, 101

zbox, 101

shapefile.Shape, 106

\_\_geo\_interface\_\_, 107

\_\_init\_\_, 107

\_\_oid, 108

\_\_repr\_\_, 107

\_errors, 108

\_from\_geojson, 107

oid, 107

parts, 108

partTypes, 108

points, 108

shapeType, 108

shapeTypeName, 107, 108

shapefile.ShapefileException, 109

shapefile.ShapeRecord, 110

\_\_geo\_interface\_\_, 110

\_\_init\_\_, 110

record, 111

shape, 111

shapefile.ShapeRecords, 112

\_\_geo\_interface\_\_, 112

\_\_repr\_\_, 112

shapefile.Shapes, 113

\_\_geo\_interface\_\_, 113

\_\_repr\_\_, 113

shapefile.Writer, 137

\_\_bbox, 140

\_\_dbfHeader, 140

\_\_dbfRecord, 140

\_\_del\_\_, 140

\_\_enter\_\_, 140

\_\_exit\_\_, 141

\_\_getFileObj, 141

\_\_init\_\_, 140

\_\_len\_\_, 141

\_\_mbox, 141

\_\_shapefileHeader, 141

\_\_shpFileLength, 141

\_\_shpRecord, 141

\_\_shxRecord, 141

\_\_zbox, 141

\_bbox, 145

\_files\_to\_close, 145

\_mbox, 145

\_shapeparts, 141

\_zbox, 145

autoBalance, 145

balance, 141

bbox, 142

close, 142

dbf, 145

deletionFlag, 145

encoding, 145

encodingErrors, 145

field, 142

fields, 145

line, 142

linem, 142

linez, 142

mbox, 142

multipatch, 142

multipoint, 143

multipointm, 143

multipointz, 143

null, 143

point, 143

pointm, 143

pointz, 143

poly, 143

polym, 144

polyz, 144

recNum, 145

record, 144

shape, 144

shapeType, 145

shapeTypeName, 144

shp, 145

shpNum, 145

shx, 145

target, 145

zbox, 144

shapeFileDir

Globals, 26

MainInputFrame.MainInput, 91

shapeGB

ShapeTest, 36

shapeLen

ShapeTest, 36

SortIntoColumns, 40

shapeMA

ShapeTest, 36

shapeName

shapefile.Reader, 101

shapeRecord

shapefile.Reader, 100

shapeRecords

shapefile.Reader, 100

shapes

shapefile.Reader, 100

ShapeTest, 36

SortIntoColumns, 40

ShapeTest, 35

areaKm2, 36

as\_dict, 36

lat, 36

lon, 36

NewSAMS, 36

pointLen, 36

record, 36

SAMS, 36

sf, 36

shapeGB, 36

shapeLen, 36

shapeMA, 36

shapes, 36

X, 36

Y, 36

ShapeTest.GeoShape, 63

\_\_init\_\_, 63

areaKm2, 63

lat, 63

lon, 63

NewSAMS, 63

SAMS, 63

X, 63

Y, 63

ShapeTest.py, 156

shapeType

shapefile.Reader, 101

shapefile.Shape, 108

shapefile.Writer, 145

SHAPETYPE\_LOOKUP

shapefile, 34

shapeTypeName

shapefile.Reader, 100

shapefile.Shape, 107, 108

shapefile.Writer, 144

shapeVal

Widgets.SubFrameInterpFunction, 134

ShowArgs

GeoSams.MainApplication, 80

ShowMessage

Globals, 23

shp

shapefile.Reader, 101

shapefile.Writer, 145

shpLength

shapefile.Reader, 101

shpNum

shapefile.Writer, 145

shx

shapefile.Reader, 101

shapefile.Writer, 145

signed\_area

shapefile, 32

simCfgDir

Globals, 26

simCfgFile

MainInputFrame.MainInput, 91

simConfigFile

GeoSams.MainApplication, 83

simStartDir

MainInputFrame.MainInput, 91

SpecialAreaFrame.SpecialArea, 130

skipStatusMsgs

GeoSams.MainApplication, 83

skipStatusMsgsRB

GeoSams.MainApplication, 83

Sort By Area Frame, 10, 12

sortAreaFrame

AreaManager.AreaManager, 50

SortByRegionFrame.SortByRegion, 126

SortByAreaFrame, 37

SortByAreaFrame.py, 157

SortByAreaFrame.SortByArea, 114

\_\_init\_\_, 117

AppendYears, 118

areaData, 119

areaFName, 119

areas, 119

browseExportButton, 119

BrowseExportFile, 118

CbUpdateUnits, 118, 119

comboParameter, 119

dataSortFileEntry, 119

dataSortFileLabel, 119

dataSortUnitsEntry, 119

dataSortUnitsLabel, 119

domainName, 119

EnterKeyClicked, 118, 119

ExportAll, 118

exportAllSortButton, 119

exportFileEntry, 119

exportFileLabel, 119

exportFileName, 119

ExportThis, 118

exportThisSortButton, 119

friend, 119

GetDataSortFile, 118

maxYears, 119

numAreas, 119

numAreasEntry, 120

numAreasLabel, 120

numAreasMax, 120

NumAreasUpdate, 118

numCorners, 120

numCornersMax, 120

numYears, 120

on\_visibility, 118, 120

openDataSortButton, 120

outputParmLabel, 120

paramStr, 120

pop\_up, 118

root, 120

RunSort, 118

runSortButton, 120

saveDataSortButton, 120

SaveDataSortFile, 118

scrollFrame, 120

startDir, 120

UpdateUnits, 118

UpdateWidgets, 118

yearStart, 120

yearStop, 120

SortByRegionFrame, 38

SortByRegionFrame.GeoShape, 64

\_\_init\_\_, 64

area, 64

lat, 64

lon, 64

X, 64

Y, 64

Zone, 64

SortByRegionFrame.py, 158

SortByRegionFrame.SortByRegion, 121

\_\_init\_\_, 123

AppendYears, 123

areaFName, 125

areaKm2, 125

browseExportButton, 125

BrowseExportFile, 123

comboParameter, 125

DetermineMgmtAreaIndex, 123

domainName, 125

ExportAll, 123

exportAllSortButton, 125

exportFileEntry, 125

exportFileLabel, 125

exportFileName, 125

ExportThis, 124

exportThisSortButton, 125

firstYrCol, 125

friend, 125

GetZoneData, 124

maxAreas, 125

maxYears, 125

numAreas, 125

numCorners, 125

numYears, 125

on\_visibility, 124, 125

outputParmLabel, 125

paramStr, 125

pop\_up, 124

resultsDir, 125

root, 125

RunSort, 124

runSortButton, 125

scrollFrame, 126

sortAreaFrame, 126

startDir, 126

table, 126

tableCols, 126

tableRows, 126

UpdateWidgets, 124

yearStart, 126

yearStop, 126

zones, 126

SortIntoColumns, 39

as\_dict, 40

columns, 40

dataFile, 40

domain, 40

fileName, 40

index, 40

inputFile, 40

l, 40

M, 40

na\_rep, 40

name, 40

nodes, 40

outfile, 40

pointLen, 40

record, 40

Region, 40

rows, 40

sep, 40

sf, 40

shape, 40

shapeFile, 40

shapeLen, 40

shapes, 40

subDir, 41

X, 41

X\_t, 41

Y, 41

Y\_t, 41

SortIntoColumns.Column, 54

\_\_init\_\_, 54

inBox, 54

name, 54

SortIntoColumns.GeoShape, 65

\_\_init\_\_, 65

lat, 65

lon, 65

Region, 65

X, 65

Y, 65

SortIntoColumns.py, 159

specAccCfgDir

Globals, 26

specAccFile

SpecialAreaFrame.SpecialArea, 130

specAccFileLabel

SpecialAreaFrame.SpecialArea, 130

specAccFileStr

GeoSams.MainApplication, 83

Special Access Area, 13

SpecialAreaFrame, 42

SpecialAreaFrame.py, 160

SpecialAreaFrame.SpecialArea, 127

\_\_init\_\_, 128

areaFName, 130

areaMgr, 130

EnterKeyClicked, 129, 130

friend, 130

GetAreaFile, 129

GetGMgrConfigFName, 129

gmCfgFile, 130

numAreas, 130

numAreasEntry, 130

numAreasLabel, 130

numAreasMax, 130

NumAreasUpdate, 129

numCorners, 130

numCornersMax, 130

on\_visibility, 129, 130

openAreaFileButton, 130

openGmgrConfigButton, 130

pop\_up, 129

root, 130

SaveAreaFile, 129

saveAreaFileButton, 130

SaveSpecialAreaData, 129

simStartDir, 130

specAccFile, 130

specAccFileLabel, 130

startDir, 130

UpdateWidgets, 129

startDayComboDay

MainInputFrame.MainInput, 91

startDayComboMonth

MainInputFrame.MainInput, 91

startDayLabel

MainInputFrame.MainInput, 91

startDir

FishMortBySpecAcc.FishMortBySpecAcc, 62

SortByAreaFrame.SortByArea, 120

SortByRegionFrame.SortByRegion, 126

SpecialAreaFrame.SpecialArea, 130

startincCol

AreaManager.AreaMgrSubFrame, 53

startYr

MainInputFrame.MainInput, 91

stopDayComboDay

MainInputFrame.MainInput, 91

stopDayComboMonth

MainInputFrame.MainInput, 91

stopDayLabel

MainInputFrame.MainInput, 92

stopYr

MainInputFrame.MainInput, 92

style

GeoSams.MainApplication, 83

subDir

SortIntoColumns, 41

surveyDataDir

Globals, 26

surveyStartDir

MainInputFrame.MainInput, 92

table

SortByRegionFrame.SortByRegion, 126

tableCols

SortByRegionFrame.SortByRegion, 126

tableRows

SortByRegionFrame.SortByRegion, 126

target

shapefile.Writer, 145

test

shapefile, 33

ToggleSkipStatusMsgs

GeoSams.MainApplication, 80

towSpdStr

GrowthFrame.Growth, 74

towSpeed

GrowthFrame.Growth, 74

TRIANGLE\_FAN

shapefile, 34

TRIANGLE\_STRIP

shapefile, 34

tsPerYear

GeoSams.MainApplication, 83

MainInputFrame.MainInput, 92

u

shapefile, 33

UpdateEntry

Globals, 24

UpdateUnits

SortByAreaFrame.SortByArea, 118

UpdateValues

GrowthFrame.Growth, 71

UpdateWidgets

AreaManager.AreaManager, 49

FishMortBySpecAcc.FishMortBySpecAcc, 61

GrowthFrame.Growth, 71

SortByAreaFrame.SortByArea, 118

SortByRegionFrame.SortByRegion, 124

SpecialAreaFrame.SpecialArea, 129

useHabCamData

GeoSams.MainApplication, 83

useMatlabRB

EditMathSetupFrame.EditMathSetup, 58

useOctaveRB

EditMathSetupFrame.EditMathSetup, 58

usingMatlab

EditMathSetupFrame.EditMathSetup, 58

VERBOSE

shapefile, 34

viewPort

Widgets.ScrollFrame, 105

Widgets, 43

floatCallback, 43

numbersCallback, 43

Widgets.py, 161

Widgets.ScrollFrame, 103

\_\_init\_\_, 104

canvas, 105

canvas\_window, 105

onEnter, 104, 105

onFrameConfigure, 104, 105

onLeave, 104, 105

onMouseWheel, 104, 105

viewPort, 105

Widgets.SubFrameElement, 131

\_\_init\_\_, 131

myEntry, 132

myLabel, 132

Widgets.SubFrameInterpFunction, 133

\_\_init\_\_, 134

dimVal, 134

funcFrame, 134

myDimRBx, 134

myDimRBy, 134

myDimRBz, 134

myShapeC, 134

myShapeG, 134

myShapeL, 134

myShapeS, 134

preconEntry, 134

preconLabel, 134

shapeVal, 134

Widgets.SubFrameXY, 135

\_\_init\_\_, 135

cornerFrame, 136

latitude, 136

longitude, 136

WriteGridMgrConfig

GeoSams.MainApplication, 80

WriteGrowthConfig

GeoSams.MainApplication, 80

WriteRecruitmentConfig

GeoSams.MainApplication, 80

WriteScallopConfig

GeoSams.MainApplication, 80

WriteSpatialFncsConfig

GeoSams.MainApplication, 81

WriteStartupFile

EditMathSetupFrame.EditMathSetup, 57

X

ShapeTest, 36

ShapeTest.GeoShape, 63

SortByRegionFrame.GeoShape, 64

SortIntoColumns, 41

SortIntoColumns.GeoShape, 65

X\_t

SortIntoColumns, 41

xrange

shapefile, 34

Y

ShapeTest, 36

ShapeTest.GeoShape, 63

SortByRegionFrame.GeoShape, 64

SortIntoColumns, 41

SortIntoColumns.GeoShape, 65

Y\_t

SortIntoColumns, 41

yearEntry

FishMortBySpecAcc.FishMortBySpecAcc, 62

yearStart

AreaManager.AreaMgrSubFrame, 53

GeoSams.MainApplication, 83

SortByAreaFrame.SortByArea, 120

SortByRegionFrame.SortByRegion, 126

yearStop

AreaManager.AreaMgrSubFrame, 53

GeoSams.MainApplication, 83

SortByAreaFrame.SortByArea, 120

SortByRegionFrame.SortByRegion, 126

zbox

shapefile.Reader, 101

shapefile.Writer, 144

Zone

SortByRegionFrame.GeoShape, 64

zones

SortByRegionFrame.SortByRegion, 126