Thomas Callaghan

Version 0.1

Contents

[GeoSAMS GUI i](#_Toc188600700)

[Thomas Callaghan i](#_Toc188600701)

[Version 0.1 i](#_Toc188600702)

[1 GeoSAMS GUI 1](#_Toc188600703)

[1.1 SHOW Args 1](#_Toc188600704)

[1.2 START Sim 1](#_Toc188600705)

[1.3 SAVE ALL Configs 1](#_Toc188600706)

[1.4 Year in file names 1](#_Toc188600707)

[2 Math Setup Frame 3](#_Toc188600708)

[2.1 Radio Button 3](#_Toc188600709)

[2.2 Write Startup File 3](#_Toc188600710)

[3 Set Fishing Mortality in Special Access Areas 4](#_Toc188600711)

[3.1 Number Defined 4](#_Toc188600712)

[3.2 Load and Save Fishing Mortality Files 4](#_Toc188600713)

[3.3 Area SubFrames 4](#_Toc188600714)

[3.3.1 Year Definitions 4](#_Toc188600715)

[3.3.2 Corners, or Fields of Defined Mortalities by Area 4](#_Toc188600716)

[4 Growth Frame 5](#_Toc188600717)

[4.1 Mortality 5](#_Toc188600718)

[4.1.1 Fishing Mortality 5](#_Toc188600719)

[4.1.2 Alpha Mortality 5](#_Toc188600720)

[4.1.3 Adult Mortality 5](#_Toc188600721)

[4.1.4 Computing Alpha 5](#_Toc188600722)

[4.1.5 Computing Natural Mortality 5](#_Toc188600723)

[4.2 Selectivity 6](#_Toc188600724)

[4.3 Incidental 6](#_Toc188600725)

[4.4 Discard 6](#_Toc188600726)

[4.5 Overall Mortality, M 6](#_Toc188600727)

[4.6 Computing Landings Per Unit Effort, LPUE 6](#_Toc188600728)

[5 Main 8](#_Toc188600729)

[5.1 Growth subframe that identifies 8](#_Toc188600730)

[5.2 Recuitment 8](#_Toc188600731)

[5.3 Configuration Files 8](#_Toc188600732)

[5.4 Output Selection 8](#_Toc188600733)

[6 Sort By Area Frame 9](#_Toc188600734)

[6.1 Number of Areas 9](#_Toc188600735)

[6.2 Output Parameters 9](#_Toc188600736)

[6.3 Load and Save Data Sort Files 9](#_Toc188600737)

[6.4 Run Sort 9](#_Toc188600738)

[6.5 Area SubFrames 9](#_Toc188600739)

[6.5.1 YYYY 9](#_Toc188600740)

[6.5.2 Comment 9](#_Toc188600741)

[6.5.3 # Corners 9](#_Toc188600742)

[6.5.4 Corner N 9](#_Toc188600743)

[7 Sort By Region Frame 11](#_Toc188600744)

[7.1 Output Parameters 11](#_Toc188600745)

[7.2 Run Sort 11](#_Toc188600746)

[8 Model Buffer, SortIntoColumns 12](#_Toc188600747)

[9 Special Access Area 13](#_Toc188600748)

[9.1 Number of Areas 13](#_Toc188600749)

[9.2 Special Access File 13](#_Toc188600750)

[9.3 Area Definitions 13](#_Toc188600751)

[9.3.1 Area N 13](#_Toc188600752)

[9.3.2 Corners: 13](#_Toc188600753)

[9.3.3 Corner N 13](#_Toc188600754)

[10 Namespace Index 14](#_Toc188600755)

[10.1 Package List 14](#_Toc188600756)

[11 Hierarchical Index 15](#_Toc188600757)

[11.1 Class Hierarchy 15](#_Toc188600758)

[12 Class Index 17](#_Toc188600759)

[12.1 Class List 17](#_Toc188600760)

[13 File Index 18](#_Toc188600761)

[13.1 File List 18](#_Toc188600762)

[14 Namespace Documentation 19](#_Toc188600763)

[14.1 AreaManager Namespace Reference 19](#_Toc188600764)

[14.1.1 Classes 19](#_Toc188600765)

[14.2 EditMathSetupFrame Namespace Reference 20](#_Toc188600766)

[14.2.1 Classes 20](#_Toc188600767)

[14.3 FishMortBySpecAcc Namespace Reference 21](#_Toc188600768)

[14.3.1 Classes 21](#_Toc188600769)

[14.4 GeoSams Namespace Reference 21](#_Toc188600770)

[14.4.1 Classes 21](#_Toc188600771)

[14.4.2 Functions 21](#_Toc188600772)

[14.4.3 Function Documentation 21](#_Toc188600773)

[14.5 Globals Namespace Reference 21](#_Toc188600774)

[14.5.1 Functions 21](#_Toc188600775)

[14.5.2 Variables 21](#_Toc188600776)

[14.5.3 Function Documentation 22](#_Toc188600777)

[14.5.4 Variable Documentation 23](#_Toc188600778)

[14.6 GrowthFrame Namespace Reference 24](#_Toc188600779)

[14.6.1 Classes 24](#_Toc188600780)

[14.7 MainInputFrame Namespace Reference 24](#_Toc188600781)

[14.7.1 Classes 24](#_Toc188600782)

[14.8 PointInPolygon Namespace Reference 24](#_Toc188600783)

[14.8.1 Functions 24](#_Toc188600784)

[14.8.2 Function Documentation 24](#_Toc188600785)

[14.9 shapefile Namespace Reference 24](#_Toc188600786)

[14.9.1 Classes 24](#_Toc188600787)

[14.9.2 Functions 25](#_Toc188600788)

[14.9.3 Variables 25](#_Toc188600789)

[14.9.4 Detailed Description 26](#_Toc188600790)

[14.9.5 Function Documentation 26](#_Toc188600791)

[14.9.6 Variable Documentation 28](#_Toc188600792)

[14.10 ShapeTest Namespace Reference 30](#_Toc188600793)

[14.10.1 Classes 30](#_Toc188600794)

[14.10.2 class GeoShapeVariables 30](#_Toc188600795)

[14.10.3 Variable Documentation 31](#_Toc188600796)

[14.11 SortByAreaFrame Namespace Reference 32](#_Toc188600797)

[14.11.1 Classes 32](#_Toc188600798)

[14.12 SortByRegionFrame Namespace Reference 32](#_Toc188600799)

[14.12.1 Classes 32](#_Toc188600800)

[14.13 SortIntoColumns Namespace Reference 32](#_Toc188600801)

[14.13.1 Classes 32](#_Toc188600802)

[14.13.2 Variables 32](#_Toc188600803)

[14.13.3 Variable Documentation 34](#_Toc188600804)

[14.14 SpecialAreaFrame Namespace Reference 35](#_Toc188600805)

[14.14.1 Classes 35](#_Toc188600806)

[14.15 Widgets Namespace Reference 35](#_Toc188600807)

[14.15.1 Classes 35](#_Toc188600808)

[14.15.2 Functions 35](#_Toc188600809)

[14.15.3 Function Documentation 35](#_Toc188600810)

[15 Class Documentation 36](#_Toc188600811)

[15.1 shapefile.\_Array Class Reference 36](#_Toc188600812)

[15.1.1 Public Member Functions 36](#_Toc188600813)

[15.1.2 Detailed Description 36](#_Toc188600814)

[15.1.3 Member Function Documentation 36](#_Toc188600815)

[15.2 shapefile.\_Record Class Reference 36](#_Toc188600816)

[15.2.1 Public Member Functions 37](#_Toc188600817)

[15.2.2 Private Attributes 37](#_Toc188600818)

[15.2.3 Detailed Description 37](#_Toc188600819)

[15.2.4 Constructor & Destructor Documentation 38](#_Toc188600820)

[15.2.5 Member Function Documentation 38](#_Toc188600821)

[15.2.6 Member Data Documentation 39](#_Toc188600822)

[15.3 AreaManager.AreaManager Class Reference 39](#_Toc188600823)

[15.3.1 Public Member Functions 40](#_Toc188600824)

[15.3.2 Public Attributes 40](#_Toc188600825)

[15.3.3 Detailed Description 41](#_Toc188600826)

[15.3.4 Constructor & Destructor Documentation 41](#_Toc188600827)

[15.3.5 Member Function Documentation 41](#_Toc188600828)

[15.3.6 Member Data Documentation 42](#_Toc188600829)

[15.4 AreaManager.AreaMgrSubFrame Class Reference 42](#_Toc188600830)

[15.4.1 Public Member Functions 43](#_Toc188600831)

[15.4.2 Public Attributes 44](#_Toc188600832)

[15.4.3 Constructor & Destructor Documentation 44](#_Toc188600833)

[15.4.4 Member Function Documentation 44](#_Toc188600834)

[15.4.5 Member Data Documentation 45](#_Toc188600835)

[15.5 SortIntoColumns.Column Class Reference 46](#_Toc188600836)

[15.5.1 Public Member Functions 46](#_Toc188600837)

[15.5.2 Public Attributes 46](#_Toc188600838)

[15.5.3 Constructor & Destructor Documentation 46](#_Toc188600839)

[15.5.4 Member Data Documentation 46](#_Toc188600840)

[15.6 AreaManager.Corner Class Reference 47](#_Toc188600841)

[15.6.1 Public Member Functions 47](#_Toc188600842)

[15.6.2 Public Attributes 47](#_Toc188600843)

[15.6.3 Detailed Description 47](#_Toc188600844)

[15.6.4 Constructor & Destructor Documentation 47](#_Toc188600845)

[15.6.5 Member Data Documentation 47](#_Toc188600846)

[15.7 EditMathSetupFrame.EditMathSetup Class Reference 48](#_Toc188600847)

[15.7.1 Public Member Functions 48](#_Toc188600848)

[15.7.2 Public Attributes 49](#_Toc188600849)

[15.7.3 Detailed Description 49](#_Toc188600850)

[15.7.4 Constructor & Destructor Documentation 49](#_Toc188600851)

[15.7.5 Member Function Documentation 49](#_Toc188600852)

[15.7.6 Member Data Documentation 50](#_Toc188600853)

[15.8 FishMortBySpecAcc.FishMortBySpecAcc Class Reference 51](#_Toc188600854)

[15.8.1 Public Member Functions 51](#_Toc188600855)

[15.8.2 Public Attributes 52](#_Toc188600856)

[15.8.3 Detailed Description 52](#_Toc188600857)

[15.8.4 Constructor & Destructor Documentation 52](#_Toc188600858)

[15.8.5 Member Function Documentation 53](#_Toc188600859)

[15.8.6 Member Data Documentation 54](#_Toc188600860)

[15.9 ShapeTest.GeoShape Class Reference 55](#_Toc188600861)

[15.9.1 Public Member Functions 55](#_Toc188600862)

[15.9.2 Public Attributes 55](#_Toc188600863)

[15.9.3 Constructor & Destructor Documentation 55](#_Toc188600864)

[15.9.4 Member Data Documentation 55](#_Toc188600865)

[15.10 SortByRegionFrame.GeoShape Class Reference 56](#_Toc188600866)

[15.10.1 Public Member Functions 56](#_Toc188600867)

[15.10.2 Public Attributes 56](#_Toc188600868)

[15.10.3 Detailed Description 56](#_Toc188600869)

[15.10.4 Constructor & Destructor Documentation 56](#_Toc188600870)

[15.10.5 Member Data Documentation 56](#_Toc188600871)

[15.11 SortIntoColumns.GeoShape Class Reference 57](#_Toc188600872)

[15.11.1 Public Member Functions 57](#_Toc188600873)

[15.11.2 Public Attributes 57](#_Toc188600874)

[15.11.3 Constructor & Destructor Documentation 57](#_Toc188600875)

[15.11.4 Member Data Documentation 57](#_Toc188600876)

[15.12 GrowthFrame.Growth Class Reference 58](#_Toc188600877)

[15.12.1 Public Member Functions 60](#_Toc188600878)

[15.12.2 Public Attributes 61](#_Toc188600879)

[15.12.3 Detailed Description 63](#_Toc188600880)

[15.12.4 Constructor & Destructor Documentation 63](#_Toc188600881)

[15.12.5 Member Function Documentation 63](#_Toc188600882)

[15.12.6 Member Data Documentation 64](#_Toc188600883)

[15.13 GeoSams.MainApplication Class Reference 67](#_Toc188600884)

[15.13.1 Public Member Functions 69](#_Toc188600885)

[15.13.2 Public Attributes 70](#_Toc188600886)

[15.13.3 Detailed Description 70](#_Toc188600887)

[15.13.4 Constructor & Destructor Documentation 70](#_Toc188600888)

[15.13.5 Member Function Documentation 71](#_Toc188600889)

[15.13.6 Member Data Documentation 74](#_Toc188600890)

[15.14 MainInputFrame.MainInput Class Reference 76](#_Toc188600891)

[15.14.1 Public Member Functions 77](#_Toc188600892)

[15.14.2 Public Attributes 78](#_Toc188600893)

[15.14.3 Detailed Description 79](#_Toc188600894)

[15.14.4 Constructor & Destructor Documentation 80](#_Toc188600895)

[15.14.5 Member Function Documentation 80](#_Toc188600896)

[15.14.6 Member Data Documentation 82](#_Toc188600897)

[15.15 shapefile.Reader Class Reference 85](#_Toc188600898)

[15.15.1 Public Member Functions 87](#_Toc188600899)

[15.15.2 Public Attributes 87](#_Toc188600900)

[15.15.3 Protected Attributes 87](#_Toc188600901)

[15.15.4 Private Member Functions 87](#_Toc188600902)

[15.15.5 Private Attributes 88](#_Toc188600903)

[15.15.6 Detailed Description 88](#_Toc188600904)

[15.15.7 Constructor & Destructor Documentation 88](#_Toc188600905)

[15.15.8 Member Function Documentation 88](#_Toc188600906)

[15.15.9 Member Data Documentation 93](#_Toc188600907)

[15.16 Widgets.ScrollFrame Class Reference 94](#_Toc188600908)

[15.16.1 Public Member Functions 94](#_Toc188600909)

[15.16.2 Public Attributes 95](#_Toc188600910)

[15.16.3 Detailed Description 95](#_Toc188600911)

[15.16.4 Constructor & Destructor Documentation 95](#_Toc188600912)

[15.16.5 Member Function Documentation 95](#_Toc188600913)

[15.16.6 Member Data Documentation 96](#_Toc188600914)

[15.17 shapefile.Shape Class Reference 96](#_Toc188600915)

[15.17.1 Public Member Functions 97](#_Toc188600916)

[15.17.2 Public Attributes 97](#_Toc188600917)

[15.17.3 Static Protected Member Functions 97](#_Toc188600918)

[15.17.4 Protected Attributes 98](#_Toc188600919)

[15.17.5 \_errorsPrivate Attributes 98](#_Toc188600920)

[15.17.6 Constructor & Destructor Documentation 98](#_Toc188600921)

[15.17.7 Member Function Documentation 98](#_Toc188600922)

[15.17.8 Member Data Documentation 99](#_Toc188600923)

[15.18 shapefile.ShapefileException Class Reference 99](#_Toc188600924)

[15.18.1 Detailed Description 99](#_Toc188600925)

[15.19 shapefile.ShapeRecord Class Reference 100](#_Toc188600926)

[15.19.1 Public Member Functions 100](#_Toc188600927)

[15.19.2 Public Attributes 100](#_Toc188600928)

[15.19.3 Detailed Description 100](#_Toc188600929)

[15.19.4 Constructor & Destructor Documentation 100](#_Toc188600930)

[15.19.5 Member Function Documentation 100](#_Toc188600931)

[15.19.6 Member Data Documentation 101](#_Toc188600932)

[15.20 shapefile.ShapeRecords Class Reference 101](#_Toc188600933)

[15.20.1 Public Member Functions 101](#_Toc188600934)

[15.20.2 Detailed Description 101](#_Toc188600935)

[15.20.3 Member Function Documentation 101](#_Toc188600936)

[15.21 shapefile.Shapes Class Reference 102](#_Toc188600937)

[15.21.1 Public Member Functions 102](#_Toc188600938)

[15.21.2 Detailed Description 102](#_Toc188600939)

[15.21.3 Member Function Documentation 102](#_Toc188600940)

[15.22 SortByAreaFrame.SortByArea Class Reference 102](#_Toc188600941)

[15.22.1 Public Member Functions 104](#_Toc188600942)

[15.22.2 Public Attributes 104](#_Toc188600943)

[15.22.3 Detailed Description 105](#_Toc188600944)

[15.22.4 Constructor & Destructor Documentation 105](#_Toc188600945)

[15.22.5 Member Function Documentation 106](#_Toc188600946)

[15.22.6 Member Data Documentation 107](#_Toc188600947)

[15.23 SortByRegionFrame.SortByRegion Class Reference 108](#_Toc188600948)

[15.23.1 Public Member Functions 110](#_Toc188600949)

[15.23.2 Public Attributes 110](#_Toc188600950)

[15.23.3 Detailed Description 111](#_Toc188600951)

[15.23.4 Constructor & Destructor Documentation 111](#_Toc188600952)

[15.23.5 Member Function Documentation 111](#_Toc188600953)

[15.23.6 Member Data Documentation 113](#_Toc188600954)

[15.24 SpecialAreaFrame.SpecialArea Class Reference 114](#_Toc188600955)

[15.24.1 Public Member Functions 115](#_Toc188600956)

[15.24.2 Public Attributes 116](#_Toc188600957)

[15.24.3 Detailed Description 116](#_Toc188600958)

[15.24.4 Constructor & Destructor Documentation 116](#_Toc188600959)

[15.24.5 Member Function Documentation 117](#_Toc188600960)

[15.24.6 Member Data Documentation 118](#_Toc188600961)

[15.25 Widgets.SubFrameElement Class Reference 119](#_Toc188600962)

[15.25.1 Public Member Functions 119](#_Toc188600963)

[15.25.2 Public Attributes 119](#_Toc188600964)

[15.25.3 Detailed Description 119](#_Toc188600965)

[15.25.4 Constructor & Destructor Documentation 119](#_Toc188600966)

[15.25.5 Member Data Documentation 120](#_Toc188600967)

[15.26 Widgets.SubFrameInterpFunction Class Reference 120](#_Toc188600968)

[15.26.1 Public Member Functions 120](#_Toc188600969)

[15.26.2 Public Attributes 120](#_Toc188600970)

[15.26.3 Constructor & Destructor Documentation 121](#_Toc188600971)

[15.26.4 Member Data Documentation 121](#_Toc188600972)

[15.27 Widgets.SubFrameXY Class Reference 122](#_Toc188600973)

[15.27.1 Public Member Functions 122](#_Toc188600974)

[15.27.2 Public Attributes 122](#_Toc188600975)

[15.27.3 Detailed Description 122](#_Toc188600976)

[15.27.4 Constructor & Destructor Documentation 122](#_Toc188600977)

[15.27.5 Member Data Documentation 123](#_Toc188600978)

[15.28 shapefile.Writer Class Reference 123](#_Toc188600979)

[15.28.1 Public Member Functions 125](#_Toc188600980)

[15.28.2 Public Attributes 125](#_Toc188600981)

[15.28.3 Protected Member Functions 125](#_Toc188600982)

[15.28.4 Protected Attributes 125](#_Toc188600983)

[15.28.5 Private Member Functions 126](#_Toc188600984)

[15.28.6 Detailed Description 126](#_Toc188600985)

[15.28.7 Constructor & Destructor Documentation 126](#_Toc188600986)

[15.28.8 Member Function Documentation 126](#_Toc188600987)

[15.28.9 Member Data Documentation 131](#_Toc188600988)

[16 File Documentation 132](#_Toc188600989)

[16.1 AreaManager.py File Reference 132](#_Toc188600990)

[16.1.1 Classes 132](#_Toc188600991)

[16.1.2 class AreaManager.AreaMgrSubFrameNamespaces 132](#_Toc188600992)

[16.2 EditMathSetupFrame.py File Reference 132](#_Toc188600993)

[16.2.1 Classes 132](#_Toc188600994)

[16.2.2 Namespaces 132](#_Toc188600995)

[16.3 FishMortBySpecAcc.py File Reference 132](#_Toc188600996)

[16.3.1 Classes 132](#_Toc188600997)

[16.3.2 Namespaces 132](#_Toc188600998)

[16.4 GeoSams.py File Reference 132](#_Toc188600999)

[16.4.1 Classes 132](#_Toc188601000)

[16.4.2 Namespaces 132](#_Toc188601001)

[16.4.3 Functions 132](#_Toc188601002)

[16.5 Globals.py File Reference 133](#_Toc188601003)

[16.5.1 Namespaces 133](#_Toc188601004)

[16.5.2 Functions 133](#_Toc188601005)

[16.5.3 Variables 133](#_Toc188601006)

[16.6 GrowthFrame.py File Reference 133](#_Toc188601007)

[16.6.1 Classes 133](#_Toc188601008)

[16.6.2 Namespaces 134](#_Toc188601009)

[16.7 MainInputFrame.py File Reference 134](#_Toc188601010)

[16.7.1 Classes 134](#_Toc188601011)

[16.7.2 Namespaces 134](#_Toc188601012)

[16.8 PointInPolygon.py File Reference 134](#_Toc188601013)

[16.8.1 Namespaces 134](#_Toc188601014)

[16.8.2 Functions 134](#_Toc188601015)

[16.9 PyshpMaster/shapefile.py File Reference 134](#_Toc188601016)

[16.9.1 Classes 134](#_Toc188601017)

[16.9.2 Namespaces 134](#_Toc188601018)

[16.9.3 Functions 134](#_Toc188601019)

[16.9.4 Variables 135](#_Toc188601020)

[16.10 ShapeTest.py File Reference 135](#_Toc188601021)

[16.10.1 Classes 135](#_Toc188601022)

[16.10.2 class ShapeTest.GeoShapeNamespaces 135](#_Toc188601023)

[16.10.3 Variables 135](#_Toc188601024)

[16.11 SortByAreaFrame.py File Reference 136](#_Toc188601025)

[16.11.1 Classes 136](#_Toc188601026)

[16.11.2 Namespaces 136](#_Toc188601027)

[16.12 SortByRegionFrame.py File Reference 136](#_Toc188601028)

[16.12.1 Classes 136](#_Toc188601029)

[16.12.2 Namespaces 136](#_Toc188601030)

[16.13 SortIntoColumns.py File Reference 136](#_Toc188601031)

[16.13.1 Classes 136](#_Toc188601032)

[16.13.2 Namespaces 136](#_Toc188601033)

[16.13.3 Variables 136](#_Toc188601034)

[16.14 SpecialAreaFrame.py File Reference 137](#_Toc188601035)

[16.14.1 Classes 137](#_Toc188601036)

[16.14.2 Namespaces 137](#_Toc188601037)

[16.15 Widgets.py File Reference 137](#_Toc188601038)

[16.15.1 Classes 137](#_Toc188601039)

[16.15.2 Namespaces 137](#_Toc188601040)

[16.15.3 Functions 137](#_Toc188601041)

[17 Index 138](#_Toc188601042)

# 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.

# Model Buffer, SortIntoColumns

For a given point, check each of the polygons in a given shape file.

If in that polygon set BUFFER\_<shape> to 1.

The region and buffer shapefiles for GB and MA are attached. **Column** "Region" for all shapefiles is the modeling region, which makes columns "REGION" and "BUFFER\_REGION" in the csvs.

Buffer is larger than the modeling region, so some data outside the modeling region but close to the boundary could be within the buffer and used to construct the models. Some data will be used several times for different models if multiple modeling regions are nearby. For example, within the blue circle in the below MA plot are a few data points outside SAMS but within the buffer, which will be used by two modeling regions. For this reason, created multiple BUFFER\_REGION columns for the survey data X\_Y\_BIOM\_AL2022\_XX\_BUFFER.csv. The output from the R code, X\_Y\_BIOM\_AL2022\_XX\_BUFFER\_GAM.csv, has a longer length than the input X\_Y\_BIOM\_AL2022\_XX\_BUFFER.csv because some data were used several times, and to make the query in the Python part easier I just duplicated some of the data. **Column** "BUFFER" in X\_Y\_BIOM\_AL2022\_XX\_BUFFER\_GAM.csv indicates which modeling region's buffer the data and GAM estimates/residuals belong to.

# 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**

**EditMathSetupFrame**

**FishMortBySpecAcc**

**GeoSams**

**Globals**

**GrowthFrame**

**MainInputFrame**

**PointInPolygon**

**shapefile**

**ShapeTest**

**SortByAreaFrame**

**SortByRegionFrame**

**SortIntoColumns**

**SpecialAreaFrame**

**Widgets**

# Hierarchical Index

## Class Hierarchy

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

array.array

shapefile.\_Array

SortIntoColumns.Column

AreaManager.Corner

Exception

shapefile.ShapefileException

tk.Frame

AreaManager.AreaMgrSubFrame

Widgets.ScrollFrame

Widgets.SubFrameElement

Widgets.SubFrameInterpFunction

Widgets.SubFrameXY

ttk.Frame

AreaManager.AreaManager

EditMathSetupFrame.EditMathSetup

FishMortBySpecAcc.FishMortBySpecAcc

GrowthFrame.Growth

MainInputFrame.MainInput

SortByAreaFrame.SortByArea

SortByRegionFrame.SortByRegion

SpecialAreaFrame.SpecialArea

ShapeTest.GeoShape

SortByRegionFrame.GeoShape

SortIntoColumns.GeoShape

list

shapefile.ShapeRecords

shapefile.Shapes

shapefile.\_Record

object

shapefile.Reader

shapefile.Shape

shapefile.ShapeRecord

shapefile.Writer

tk.Tk

GeoSams.MainApplication

GrowthFrame.Growth

# Class Index

## Class List

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

**shapefile.\_Array**

**shapefile.\_Record**

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

**AreaManager.AreaMgrSubFrame**

**SortIntoColumns.Column**

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

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

**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 )**

**ShapeTest.GeoShape**

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

**SortIntoColumns.GeoShape**

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

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

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

**shapefile.Reader**

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

**shapefile.Shape**

**shapefile.ShapefileException**

**shapefile.ShapeRecord**

**shapefile.ShapeRecords**

**shapefile.Shapes**

**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 )**

**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 )**

**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 )**

**Widgets.SubFrameElement (Generic Element )**

**Widgets.SubFrameInterpFunction**

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

**shapefile.Writer**

# File Index

## File List

Here is a list of all files with brief descriptions:

**AreaManager.py**

**EditMathSetupFrame.py**

**FishMortBySpecAcc.py**

**GeoSams.py**

**Globals.py**

**GrowthFrame.py**

**MainInputFrame.py**

**PointInPolygon.py**

**ShapeTest.py**

**SortByAreaFrame.py**

**SortByRegionFrame.py**

**SortIntoColumns.py**

**SpecialAreaFrame.py**

**Widgets.py**

**PyshpMaster/shapefile.py**

# 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.70*).

### 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.71*).

#### 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.63*).

### 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.63*).

#### 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, 126

\_\_dbfHdrLength

shapefile.Reader, 93

\_\_dbfHeader

shapefile.Reader, 88

shapefile.Writer, 126

\_\_dbfRecord

shapefile.Writer, 126

\_\_del\_\_

shapefile.Reader, 88

shapefile.Writer, 126

\_\_dir\_\_

shapefile.\_Record, 38

\_\_enter\_\_

shapefile.Reader, 88

shapefile.Writer, 126

\_\_exit\_\_

shapefile.Reader, 89

shapefile.Writer, 126

\_\_field\_positions

shapefile.\_Record, 39

\_\_fieldLookup

shapefile.Reader, 93

\_\_fullRecLookup

shapefile.Reader, 93

\_\_fullRecStruct

shapefile.Reader, 93

\_\_geo\_interface\_\_

shapefile.Reader, 89

shapefile.Shape, 98

shapefile.ShapeRecord, 100

shapefile.ShapeRecords, 101

shapefile.Shapes, 102

\_\_getattr\_\_

shapefile.\_Record, 38

\_\_getFileObj

shapefile.Reader, 89

shapefile.Writer, 127

\_\_getitem\_\_

shapefile.\_Record, 38

\_\_init\_\_

AreaManager.AreaManager, 41

AreaManager.AreaMgrSubFrame, 44

AreaManager.Corner, 47

EditMathSetupFrame.EditMathSetup, 49

FishMortBySpecAcc.FishMortBySpecAcc, 52

GeoSams.MainApplication, 70

GrowthFrame.Growth, 63

MainInputFrame.MainInput, 80

shapefile.\_Record, 38

shapefile.Reader, 88

shapefile.Shape, 98

shapefile.ShapeRecord, 100

shapefile.Writer, 126

ShapeTest.GeoShape, 55

SortByAreaFrame.SortByArea, 105

SortByRegionFrame.GeoShape, 56

SortByRegionFrame.SortByRegion, 111

SortIntoColumns.Column, 46

SortIntoColumns.GeoShape, 57

SpecialAreaFrame.SpecialArea, 116

Widgets.ScrollFrame, 95

Widgets.SubFrameElement, 119

Widgets.SubFrameInterpFunction, 121

Widgets.SubFrameXY, 122

\_\_iter\_\_

shapefile.Reader, 89

\_\_len\_\_

shapefile.Reader, 89

shapefile.Writer, 127

\_\_mbox

shapefile.Writer, 127

\_\_oid

shapefile.\_Record, 39

shapefile.Shape, 99

\_\_record

shapefile.Reader, 89

\_\_recordFields

shapefile.Reader, 89

\_\_recordFmt

shapefile.Reader, 89

\_\_recordLength

shapefile.Reader, 93

\_\_repr\_\_

shapefile.\_Array, 36

shapefile.\_Record, 39

shapefile.Shape, 98

shapefile.ShapeRecords, 101

shapefile.Shapes, 102

\_\_restrictIndex

shapefile.Reader, 89

\_\_setattr\_\_

shapefile.\_Record, 39

\_\_setitem\_\_

shapefile.\_Record, 39

\_\_shape

shapefile.Reader, 90

\_\_shapefileHeader

shapefile.Writer, 127

\_\_shapeIndex

shapefile.Reader, 90

\_\_shpFileLength

shapefile.Writer, 127

\_\_shpHeader

shapefile.Reader, 90

\_\_shpRecord

shapefile.Writer, 127

\_\_shxHeader

shapefile.Reader, 90

\_\_shxOffsets

shapefile.Reader, 90

\_\_shxRecord

shapefile.Writer, 127

\_\_str\_\_

shapefile.Reader, 90

\_\_version\_\_

shapefile, 28

\_\_zbox

shapefile.Writer, 127

\_bbox

shapefile.Writer, 131

\_errors

shapefile.Shape, 99

\_files\_to\_close

shapefile.Reader, 93

shapefile.Writer, 131

\_from\_geojson

shapefile.Shape, 98

\_mbox

shapefile.Writer, 131

\_offsets

shapefile.Reader, 93

\_shapeparts

shapefile.Writer, 127

\_zbox

shapefile.Writer, 131

ABUN

Globals, 23

abunVar

MainInputFrame.MainInput, 82

addFrameClicked

GeoSams.MainApplication, 74

alphaMort

GrowthFrame.Growth, 64

alphStr

GrowthFrame.Growth, 64

analDir

Globals, 23

AppendResults

AreaManager.AreaMgrSubFrame, 44

AppendYears

SortByAreaFrame.SortByArea, 106

SortByRegionFrame.SortByRegion, 111

area

SortByRegionFrame.GeoShape, 56

areaData

AreaManager.AreaManager, 42

SortByAreaFrame.SortByArea, 107

areaFName

MainInputFrame.MainInput, 82

SortByAreaFrame.SortByArea, 107

SortByRegionFrame.SortByRegion, 113

SpecialAreaFrame.SpecialArea, 118

areaFrame

AreaManager.AreaMgrSubFrame, 45

areaKm2

ShapeTest, 31

ShapeTest.GeoShape, 55

SortByRegionFrame.SortByRegion, 113

AreaManager, 19

AreaManager.AreaManager, 39

\_\_init\_\_, 41

areaData, 42

areaSubFrame, 42

cornerRow, 42

numAreas, 42

numAreasMax, 42

NumAreasUpdate, 41

numCorners, 42

numCornersMax, 42

ReadAreaCorners, 41

scrollFrame, 42

sortAreaFrame, 42

UpdateWidgets, 42

AreaManager.AreaMgrSubFrame, 42

\_\_init\_\_, 44

AppendResults, 44

areaFrame, 45

comment, 45

commentEntry, 45

compAreaEntry, 45

corners, 45

EnterKeyClicked, 44

myArea, 45

numCorners, 45

numCornersEntry, 45

numCornersMax, 45

NumCornersUpdate, 44

numYearsMax, 45

numYrCols, 45

results, 45

SaveField, 44

startincCol, 45

yearStart, 45

yearStop, 45

AreaManager.Corner, 47

\_\_init\_\_, 47

lat, 47

long, 47

numCorners, 47

AreaManager.py, 132

areaMgr

FishMortBySpecAcc.FishMortBySpecAcc, 54

SpecialAreaFrame.SpecialArea, 118

areas

SortByAreaFrame.SortByArea, 107

areaSubFrame

AreaManager.AreaManager, 42

as\_dict

shapefile.\_Record, 39

ShapeTest, 31

SortIntoColumns, 34

autoBalance

shapefile.Writer, 131

b

shapefile, 26

balance

shapefile.Writer, 127

bbox

shapefile.Reader, 93

shapefile.Writer, 128

bbox\_contains

shapefile, 26

bbox\_overlap

shapefile, 26

BIOM

Globals, 23

bmsVar

MainInputFrame.MainInput, 82

browseExportButton

SortByAreaFrame.SortByArea, 107

SortByRegionFrame.SortByRegion, 113

BrowseExportFile

SortByAreaFrame.SortByArea, 106

SortByRegionFrame.SortByRegion, 111

canvas

Widgets.ScrollFrame, 96

canvas\_window

Widgets.ScrollFrame, 96

CBSelectedOutput

MainInputFrame.MainInput, 80

CbUpdateUnits

SortByAreaFrame.SortByArea, 106, 107

CheckStartDay

MainInputFrame.MainInput, 80, 82

CheckStopDay

MainInputFrame.MainInput, 80, 82

close

shapefile.Reader, 90

shapefile.Writer, 128

columns

SortIntoColumns, 34

comboParameter

SortByAreaFrame.SortByArea, 107

SortByRegionFrame.SortByRegion, 113

comboTFStr

Globals, 23

comment

AreaManager.AreaMgrSubFrame, 45

commentEntry

AreaManager.AreaMgrSubFrame, 45

compAreaEntry

AreaManager.AreaMgrSubFrame, 45

ComputeResiduals

GeoSams, 21

ComputeSelectOuputValue

MainInputFrame.MainInput, 80

configDir

Globals, 23

ConvertDayOfYrToMonthDay

GeoSams.MainApplication, 71

ConvertMonthDayToDayOfYr

GeoSams.MainApplication, 71

cornerFrame

Widgets.SubFrameXY, 123

cornerLabelArr

Globals, 23

cornerRow

AreaManager.AreaManager, 42

corners

AreaManager.AreaMgrSubFrame, 45

dataDir

Globals, 23

dataFile

SortIntoColumns, 34

dataSortFileEntry

SortByAreaFrame.SortByArea, 107

dataSortFileLabel

SortByAreaFrame.SortByArea, 107

dataSortUnitsEntry

SortByAreaFrame.SortByArea, 107

dataSortUnitsLabel

SortByAreaFrame.SortByArea, 107

daysInYear

GeoSams.MainApplication, 74

dbf

shapefile.Reader, 93

shapefile.Writer, 131

deletionFlag

shapefile.Writer, 131

desiredOutput

MainInputFrame.MainInput, 82

DetermineMgmtAreaIndex

SortByRegionFrame.SortByRegion, 111

DetermineUnitsScale

Globals, 22

dimVal

Widgets.SubFrameInterpFunction, 121

domain

SortIntoColumns, 34

domainName

GeoSams.MainApplication, 74

SortByAreaFrame.SortByArea, 107

SortByRegionFrame.SortByRegion, 113

domainNameCombo

MainInputFrame.MainInput, 82

domainNameLabel

MainInputFrame.MainInput, 82

dredgeDataFile

MainInputFrame.MainInput, 82

dredgeWdStr

GrowthFrame.Growth, 64

dredgeWth

GrowthFrame.Growth, 64

EBMS

Globals, 23

ebmsVar

MainInputFrame.MainInput, 82

editMathFrame

EditMathSetupFrame.EditMathSetup, 50

EditMathSetupFrame, 20

EditMathSetupFrame.EditMathSetup, 48

\_\_init\_\_, 49

editMathFrame, 50

editText, 50

editTextButton, 50

isMatlab, 50

LoadStartupFile, 49

matlabFName, 50

octaveFName, 50

on\_visibility, 49, 50

pop\_up, 49

ReadFile, 49

useMatlabRB, 50

useOctaveRB, 50

usingMatlab, 50

WriteStartupFile, 49

EditMathSetupFrame.py, 132

editText

EditMathSetupFrame.EditMathSetup, 50

editTextButton

EditMathSetupFrame.EditMathSetup, 50

encoding

shapefile.Reader, 93

shapefile.Writer, 131

encodingErrors

shapefile.Reader, 93

shapefile.Writer, 131

EnterKeyClicked

AreaManager.AreaMgrSubFrame, 44

FishMortBySpecAcc.FishMortBySpecAcc, 53, 54

MainInputFrame.MainInput, 80

SortByAreaFrame.SortByArea, 106, 107

SpecialAreaFrame.SpecialArea, 117, 118

ExportAll

SortByAreaFrame.SortByArea, 106

SortByRegionFrame.SortByRegion, 111

exportAllSortButton

SortByAreaFrame.SortByArea, 107

SortByRegionFrame.SortByRegion, 113

exportFileEntry

SortByAreaFrame.SortByArea, 107

SortByRegionFrame.SortByRegion, 113

exportFileLabel

SortByAreaFrame.SortByArea, 107

SortByRegionFrame.SortByRegion, 113

exportFileName

SortByAreaFrame.SortByArea, 107

SortByRegionFrame.SortByRegion, 113

ExportThis

SortByAreaFrame.SortByArea, 106

SortByRegionFrame.SortByRegion, 111

exportThisSortButton

SortByAreaFrame.SortByArea, 107

SortByRegionFrame.SortByRegion, 113

failure\_count

shapefile, 28

FEFF

Globals, 23

feffVar

MainInputFrame.MainInput, 82

field

shapefile.Writer, 128

fields

shapefile.Reader, 93

shapefile.Writer, 131

fileName

SortIntoColumns, 34

FIRST\_RING

shapefile, 28

firstYrCol

SortByRegionFrame.SortByRegion, 113

fishMort

GrowthFrame.Growth, 64

FishMortBySpecAcc, 21

FishMortBySpecAcc.FishMortBySpecAcc, 51

\_\_init\_\_, 52

areaMgr, 54

EnterKeyClicked, 53, 54

fishMortFile, 54

fishMortFileLabel, 54

fmFName, 54

GetFMFile, 53

numAreas, 54

numDefined, 54

numDefinedEntry, 54

numDefinedLabel, 54

numDefinedMax, 54

NumDefinedUpdate, 53

numFieldsMax, 54

numFieldss, 54

on\_visibility, 53, 54

openFMFileButton, 54

pop\_up, 53

ReadFields, 53

root, 54

SaveFMFile, 53

saveFMFileButton, 54

startDir, 54

UpdateWidgets, 53

yearEntry, 54

FishMortBySpecAcc.py, 132

fishMortFile

FishMortBySpecAcc.FishMortBySpecAcc, 54

fishMortFileLabel

FishMortBySpecAcc.FishMortBySpecAcc, 54

floatCallback

Widgets, 35

fmFName

FishMortBySpecAcc.FishMortBySpecAcc, 54

FMOR

Globals, 23

fmortFileName

GrowthFrame.Growth, 64

fmortStr

GrowthFrame.Growth, 64

fmortVar

MainInputFrame.MainInput, 82

frame1

GeoSams.MainApplication, 74

frame2

GeoSams.MainApplication, 74

frame3

GeoSams.MainApplication, 74

frame5

GeoSams.MainApplication, 74

frame6

GeoSams.MainApplication, 74

frame7

GeoSams.MainApplication, 74

frame8

GeoSams.MainApplication, 74

frameHeight

Globals, 23

frameWidth

Globals, 23

friend

GrowthFrame.Growth, 64

MainInputFrame.MainInput, 82

SortByAreaFrame.SortByArea, 107

SortByRegionFrame.SortByRegion, 113

SpecialAreaFrame.SpecialArea, 118

funcFrame

Widgets.SubFrameInterpFunction, 121

gbAdultMort

GrowthFrame.Growth, 64

gbAdultMortStr

GrowthFrame.Growth, 64

gbClFSelAStr

GrowthFrame.Growth, 64

gbClFSelBStr

GrowthFrame.Growth, 64

gbClosedFSelectA

GrowthFrame.Growth, 64

gbClosedFSelectB

GrowthFrame.Growth, 64

gbCullSize

GrowthFrame.Growth, 64

gbCullStr

GrowthFrame.Growth, 64

gbDiscard

GrowthFrame.Growth, 64

gbDiscStr

GrowthFrame.Growth, 64

gbIncident

GrowthFrame.Growth, 64

gbIncidStr

GrowthFrame.Growth, 64

gbLen0Str

GrowthFrame.Growth, 64

gbLength0

GrowthFrame.Growth, 64

gbOpenFSelectA

GrowthFrame.Growth, 64

gbOpenFSelectB

GrowthFrame.Growth, 65

gbOpFSelAStr

GrowthFrame.Growth, 65

gbOpFSelBStr

GrowthFrame.Growth, 65

gbShapeBufferFileEntry

MainInputFrame.MainInput, 82

gbShapeFileEntry

MainInputFrame.MainInput, 82

geometryStr

Globals, 23

GeoSams, 21

ComputeResiduals, 21

main, 21

GeoSAMS GUI, 1

GeoSams.MainApplication, 67

\_\_init\_\_, 70

addFrameClicked, 74

ConvertDayOfYrToMonthDay, 71

ConvertMonthDayToDayOfYr, 71

daysInYear, 74

domainName, 74

frame1, 74

frame2, 74

frame3, 74

frame5, 74

frame6, 74

frame7, 74

frame8, 74

gmConfigFile, 74

InterpAndPlotResults, 71

isSkip, 74

maxAreas, 74

maxCorners, 74

maxYears, 74

monDict, 74

notebook, 74

paramStr, 74

paramVal, 74

pop\_up, 71

ReadConfigFile, 71

ReadGridMgrConfigFile, 72

ReadRecruitConfigFile, 72

ReadSimConfigFile, 72

recrConfigFile, 74

recruitYrStop, 74

recruitYrStrt, 74

root, 74

Run\_Sim, 72

SaveConfigFiles, 72

ShowArgs, 72

simConfigFile, 75

skipStatusMsgs, 75

skipStatusMsgsRB, 75

specAccFileStr, 75

style, 75

ToggleSkipStatusMsgs, 72

tsPerYear, 75

useHabCamData, 75

WriteGridMgrConfig, 72

WriteGrowthConfig, 72

WriteRecruitmentConfig, 72

WriteScallopConfig, 72

WriteSpatialFncsConfig, 73

yearStart, 75

yearStop, 75

GeoSams.py, 132

GetAreaFile

SpecialAreaFrame.SpecialArea, 117

GetDataSortFile

SortByAreaFrame.SortByArea, 106

GetFMFile

FishMortBySpecAcc.FishMortBySpecAcc, 53

GetGMgrConfigFName

SpecialAreaFrame.SpecialArea, 117

GetGrowthConfigFName

GrowthFrame.Growth, 63

GetRecrConfigFName

MainInputFrame.MainInput, 80

GetSelectedOutputs

MainInputFrame.MainInput, 80

GetSimConfigFName

MainInputFrame.MainInput, 80

GetZoneData

SortByRegionFrame.SortByRegion, 112

Globals, 21

ABUN, 23

analDir, 23

BIOM, 23

comboTFStr, 23

configDir, 23

cornerLabelArr, 23

dataDir, 23

DetermineUnitsScale, 22

EBMS, 23

FEFF, 23

FMOR, 23

frameHeight, 23

frameWidth, 23

geometryStr, 23

grid\_area\_sqm, 23

gridDir, 23

helpXoffset, 23

helpYoffset, 23

interCfgDir, 23

LAND, 23

LNDW, 23

LPUE, 23

meters\_per\_naut\_mile, 23

RECR, 23

resultsDir, 24

scrollFrameHeight, 24

scrollFrameWidth, 24

shapeFileDir, 24

ShowMessage, 22

simCfgDir, 24

specAccCfgDir, 24

surveyDataDir, 24

UpdateEntry, 22

Globals.py, 133

gmCfgFile

SpecialAreaFrame.SpecialArea, 118

gmConfigFile

GeoSams.MainApplication, 74

grid\_area\_sqm

Globals, 23

gridDir

Globals, 23

Growth Frame, 5

growthCfgFile

GrowthFrame.Growth, 65

GrowthFrame, 24

GrowthFrame.Growth, 58

\_\_init\_\_, 63

alphaMort, 64

alphStr, 64

dredgeWdStr, 64

dredgeWth, 64

fishMort, 64

fmortFileName, 64

fmortStr, 64

friend, 64

gbAdultMort, 64

gbAdultMortStr, 64

gbClFSelAStr, 64

gbClFSelBStr, 64

gbClosedFSelectA, 64

gbClosedFSelectB, 64

gbCullSize, 64

gbCullStr, 64

gbDiscard, 64

gbDiscStr, 64

gbIncident, 64

gbIncidStr, 64

gbLen0Str, 64

gbLength0, 64

gbOpenFSelectA, 64

gbOpenFSelectB, 65

gbOpFSelAStr, 65

gbOpFSelBStr, 65

GetGrowthConfigFName, 63

growthCfgFile, 65

growthStartDir, 65

loadGrowthConfigButton, 65

LoadGrowthData, 63

lpueIntcept, 65

lpueIntcStr, 65

lpueSl2Str, 65

lpueSlope, 65

lpueSlope2, 65

lpueSlStr, 65

maAdultMort, 65

maAdultMortStr, 65

maCullSize, 65

maCullStr, 65

maDiscard, 65

maDiscStr, 65

maFSelAStr, 65

maFSelBStr, 65

maFSelectA, 65

maFSelectB, 65

maIncident, 65

maIncidStr, 65

maLen0Str, 65

maLength0, 66

maxPerDay, 66

maxPerDayStr, 66

maxTime, 66

maxTimeStr, 66

on\_visibility, 63, 66

pop\_up, 63

root, 66

saveGrowthConfigButton, 66

towSpdStr, 66

towSpeed, 66

UpdateValues, 63

UpdateWidgets, 63

GrowthFrame.py, 133

growthStartDir

GrowthFrame.Growth, 65

habCamDataFile

MainInputFrame.MainInput, 82

helpXoffset

Globals, 23

helpYoffset

Globals, 23

inBox

SortIntoColumns.Column, 46

index

SortIntoColumns, 34

INNER\_RING

shapefile, 28

inputFile

SortIntoColumns, 34

interCfgDir

Globals, 23

InterpAndPlotResults

GeoSams.MainApplication, 71

interpStartDir

MainInputFrame.MainInput, 82

is\_cw

shapefile, 26

is\_string

shapefile, 26

isMatlab

EditMathSetupFrame.EditMathSetup, 50

isSkip

GeoSams.MainApplication, 74

iterRecords

shapefile.Reader, 90

iterShapeRecords

shapefile.Reader, 90

iterShapes

shapefile.Reader, 91

izip

shapefile, 28

l

SortIntoColumns, 34

LAND

Globals, 23

landVar

MainInputFrame.MainInput, 82

lat

AreaManager.Corner, 47

ShapeTest, 31

ShapeTest.GeoShape, 55

SortByRegionFrame.GeoShape, 56

SortIntoColumns.GeoShape, 57

latitude

Widgets.SubFrameXY, 123

line

shapefile.Writer, 128

linem

shapefile.Writer, 128

linez

shapefile.Writer, 128

LNDW

Globals, 23

lndwVar

MainInputFrame.MainInput, 82

load

shapefile.Reader, 91

load\_dbf

shapefile.Reader, 91

load\_shp

shapefile.Reader, 91

load\_shx

shapefile.Reader, 91

loadGrowthConfigButton

GrowthFrame.Growth, 65

LoadGrowthData

GrowthFrame.Growth, 63

LoadStartupFile

EditMathSetupFrame.EditMathSetup, 49

logger

shapefile, 28

lon

ShapeTest, 31

ShapeTest.GeoShape, 55

SortByRegionFrame.GeoShape, 56

SortIntoColumns.GeoShape, 57

long

AreaManager.Corner, 47

longitude

Widgets.SubFrameXY, 123

LPUE

Globals, 23

lpueIntcept

GrowthFrame.Growth, 65

lpueIntcStr

GrowthFrame.Growth, 65

lpueSl2Str

GrowthFrame.Growth, 65

lpueSlope

GrowthFrame.Growth, 65

lpueSlope2

GrowthFrame.Growth, 65

lpueSlStr

GrowthFrame.Growth, 65

lpueVar

MainInputFrame.MainInput, 82

M

SortIntoColumns, 34

maAdultMort

GrowthFrame.Growth, 65

maAdultMortStr

GrowthFrame.Growth, 65

maCullSize

GrowthFrame.Growth, 65

maCullStr

GrowthFrame.Growth, 65

maDiscard

GrowthFrame.Growth, 65

maDiscStr

GrowthFrame.Growth, 65

maFSelAStr

GrowthFrame.Growth, 65

maFSelBStr

GrowthFrame.Growth, 65

maFSelectA

GrowthFrame.Growth, 65

maFSelectB

GrowthFrame.Growth, 65

main

GeoSams, 21

Main, 8

maIncident

GrowthFrame.Growth, 65

maIncidStr

GrowthFrame.Growth, 65

MainInputFrame, 24

MainInputFrame.MainInput, 76

\_\_init\_\_, 80

abunVar, 82

areaFName, 82

bmsVar, 82

CBSelectedOutput, 80

CheckStartDay, 80, 82

CheckStopDay, 80, 82

ComputeSelectOuputValue, 80

desiredOutput, 82

domainNameCombo, 82

domainNameLabel, 82

dredgeDataFile, 82

ebmsVar, 82

EnterKeyClicked, 80

feffVar, 82

fmortVar, 82

friend, 82

gbShapeBufferFileEntry, 82

gbShapeFileEntry, 82

GetRecrConfigFName, 80

GetSelectedOutputs, 80

GetSimConfigFName, 80

habCamDataFile, 82

interpStartDir, 82

landVar, 82

lndwVar, 82

lpueVar, 82

maShapeBufferFileEntry, 82

maShapeFileEntry, 82

maxYears, 82

monthsArr, 83

numYrsAvg, 83

OpenPDF, 81

openPDFButton, 83

openRecrConfigButton, 83

openSimConfigButton, 83

pop\_up, 81

recrCfgFile, 83

recrVar, 83

recrYrStop, 83

recrYrStrt, 83

root, 83

setDredgeDataButton, 83

SetDredgeFileEnvVar, 81

SetDredgeFileName, 81

SetGbShapeBufferFile, 81

setGbShapeBufferFileButton, 83

SetGBShapeBufferFileEnvVar, 81

SetGbShapeFile, 81

setGbShapeFileButton, 83

SetGBShapeFileEnvVar, 81

setHabCamDataButton, 83

SetHabCamFileEnvVar, 81

SetHabCamFileName, 81

SetMaShapeBufferFile, 81

setMaShapeBufferFileButton, 83

SetMaShapeBufferFileEnvVar, 81

SetMaShapeFile, 81

setMaShapeFileButton, 83

SetMaShapeFileEnvVar, 81

shapeFileDir, 83

simCfgFile, 83

simStartDir, 83

startDayComboDay, 83

startDayComboMonth, 83

startDayLabel, 83

startYr, 83

stopDayComboDay, 83

stopDayComboMonth, 83

stopDayLabel, 84

stopYr, 84

surveyStartDir, 84

tsPerYear, 84

MainInputFrame.py, 134

maLen0Str

GrowthFrame.Growth, 65

maLength0

GrowthFrame.Growth, 66

maShapeBufferFileEntry

MainInputFrame.MainInput, 82

maShapeFileEntry

MainInputFrame.MainInput, 82

Math Setup Frame, 3

matlabFName

EditMathSetupFrame.EditMathSetup, 50

maxAreas

GeoSams.MainApplication, 74

SortByRegionFrame.SortByRegion, 113

maxCorners

GeoSams.MainApplication, 74

maxPerDay

GrowthFrame.Growth, 66

maxPerDayStr

GrowthFrame.Growth, 66

maxTime

GrowthFrame.Growth, 66

maxTimeStr

GrowthFrame.Growth, 66

maxYears

GeoSams.MainApplication, 74

MainInputFrame.MainInput, 82

SortByAreaFrame.SortByArea, 107

SortByRegionFrame.SortByRegion, 113

mbox

shapefile.Reader, 93

shapefile.Writer, 128

meters\_per\_naut\_mile

Globals, 23

MISSING

shapefile, 28

Model Buffer, SortIntoColumns, 12

monDict

GeoSams.MainApplication, 74

monthsArr

MainInputFrame.MainInput, 83

multipatch

shapefile.Writer, 128

MULTIPATCH

shapefile, 28

multipoint

shapefile.Writer, 129

MULTIPOINT

shapefile, 28

multipointm

shapefile.Writer, 129

MULTIPOINTM

shapefile, 28

multipointz

shapefile.Writer, 129

MULTIPOINTZ

shapefile, 28

myArea

AreaManager.AreaMgrSubFrame, 45

myDimRBx

Widgets.SubFrameInterpFunction, 121

myDimRBy

Widgets.SubFrameInterpFunction, 121

myDimRBz

Widgets.SubFrameInterpFunction, 121

myEntry

Widgets.SubFrameElement, 120

myLabel

Widgets.SubFrameElement, 120

myShapeC

Widgets.SubFrameInterpFunction, 121

myShapeG

Widgets.SubFrameInterpFunction, 121

myShapeL

Widgets.SubFrameInterpFunction, 121

myShapeS

Widgets.SubFrameInterpFunction, 121

na\_rep

SortIntoColumns, 34

name

SortIntoColumns, 34

SortIntoColumns.Column, 46

NewSAMS

ShapeTest, 31

ShapeTest.GeoShape, 55

NODATA

shapefile, 28

nodes

SortIntoColumns, 34

notebook

GeoSams.MainApplication, 74

null

shapefile.Writer, 129

NULL

shapefile, 28

numAreas

AreaManager.AreaManager, 42

FishMortBySpecAcc.FishMortBySpecAcc, 54

SortByAreaFrame.SortByArea, 107

SortByRegionFrame.SortByRegion, 113

SpecialAreaFrame.SpecialArea, 118

numAreasEntry

SortByAreaFrame.SortByArea, 108

SpecialAreaFrame.SpecialArea, 118

numAreasLabel

SortByAreaFrame.SortByArea, 108

SpecialAreaFrame.SpecialArea, 118

numAreasMax

AreaManager.AreaManager, 42

SortByAreaFrame.SortByArea, 108

SpecialAreaFrame.SpecialArea, 118

NumAreasUpdate

AreaManager.AreaManager, 41

SortByAreaFrame.SortByArea, 106

SpecialAreaFrame.SpecialArea, 117

numbersCallback

Widgets, 35

numCorners

AreaManager.AreaManager, 42

AreaManager.AreaMgrSubFrame, 45

AreaManager.Corner, 47

SortByAreaFrame.SortByArea, 108

SortByRegionFrame.SortByRegion, 113

SpecialAreaFrame.SpecialArea, 118

numCornersEntry

AreaManager.AreaMgrSubFrame, 45

numCornersMax

AreaManager.AreaManager, 42

AreaManager.AreaMgrSubFrame, 45

SortByAreaFrame.SortByArea, 108

SpecialAreaFrame.SpecialArea, 118

NumCornersUpdate

AreaManager.AreaMgrSubFrame, 44

numDefined

FishMortBySpecAcc.FishMortBySpecAcc, 54

numDefinedEntry

FishMortBySpecAcc.FishMortBySpecAcc, 54

numDefinedLabel

FishMortBySpecAcc.FishMortBySpecAcc, 54

numDefinedMax

FishMortBySpecAcc.FishMortBySpecAcc, 54

NumDefinedUpdate

FishMortBySpecAcc.FishMortBySpecAcc, 53

numFieldsMax

FishMortBySpecAcc.FishMortBySpecAcc, 54

numFieldss

FishMortBySpecAcc.FishMortBySpecAcc, 54

numRecords

shapefile.Reader, 93

numShapes

shapefile.Reader, 93

numYears

SortByAreaFrame.SortByArea, 108

SortByRegionFrame.SortByRegion, 113

numYearsMax

AreaManager.AreaMgrSubFrame, 45

numYrCols

AreaManager.AreaMgrSubFrame, 45

numYrsAvg

MainInputFrame.MainInput, 83

octaveFName

EditMathSetupFrame.EditMathSetup, 50

oid

shapefile.\_Record, 39

shapefile.Shape, 98

on\_visibility

EditMathSetupFrame.EditMathSetup, 49, 50

FishMortBySpecAcc.FishMortBySpecAcc, 53, 54

GrowthFrame.Growth, 63, 66

SortByAreaFrame.SortByArea, 106, 108

SortByRegionFrame.SortByRegion, 112, 113

SpecialAreaFrame.SpecialArea, 117, 118

onEnter

Widgets.ScrollFrame, 95, 96

onFrameConfigure

Widgets.ScrollFrame, 95, 96

onLeave

Widgets.ScrollFrame, 95, 96

onMouseWheel

Widgets.ScrollFrame, 95, 96

openAreaFileButton

SpecialAreaFrame.SpecialArea, 118

openDataSortButton

SortByAreaFrame.SortByArea, 108

openFMFileButton

FishMortBySpecAcc.FishMortBySpecAcc, 54

openGmgrConfigButton

SpecialAreaFrame.SpecialArea, 118

OpenPDF

MainInputFrame.MainInput, 81

openPDFButton

MainInputFrame.MainInput, 83

openRecrConfigButton

MainInputFrame.MainInput, 83

openSimConfigButton

MainInputFrame.MainInput, 83

organize\_polygon\_rings

shapefile, 26

OUTER\_RING

shapefile, 28

outfile

SortIntoColumns, 34

outputParmLabel

SortByAreaFrame.SortByArea, 108

SortByRegionFrame.SortByRegion, 113

paramStr

GeoSams.MainApplication, 74

SortByAreaFrame.SortByArea, 108

SortByRegionFrame.SortByRegion, 113

paramVal

GeoSams.MainApplication, 74

parts

shapefile.Shape, 99

PARTTYPE\_LOOKUP

shapefile, 28

partTypes

shapefile.Shape, 99

pathlike\_obj

shapefile, 27

point

shapefile.Writer, 129

POINT

shapefile, 29

PointInPolygon, 24

PointInPolygon, 24

PointInPolygon.py, 134

pointLen

ShapeTest, 31

SortIntoColumns, 34

pointm

shapefile.Writer, 129

POINTM

shapefile, 29

points

shapefile.Shape, 99

pointz

shapefile.Writer, 129

POINTZ

shapefile, 29

poly

shapefile.Writer, 129

POLYGON

shapefile, 29

POLYGONM

shapefile, 29

POLYGONZ

shapefile, 29

POLYLINE

shapefile, 29

POLYLINEM

shapefile, 29

POLYLINEZ

shapefile, 29

polym

shapefile.Writer, 129

polyz

shapefile.Writer, 130

pop\_up

EditMathSetupFrame.EditMathSetup, 49

FishMortBySpecAcc.FishMortBySpecAcc, 53

GeoSams.MainApplication, 71

GrowthFrame.Growth, 63

MainInputFrame.MainInput, 81

SortByAreaFrame.SortByArea, 106

SortByRegionFrame.SortByRegion, 112

SpecialAreaFrame.SpecialArea, 117

preconEntry

Widgets.SubFrameInterpFunction, 121

preconLabel

Widgets.SubFrameInterpFunction, 121

PyshpMaster/shapefile.py, 134

PYTHON3

shapefile, 29

ReadAreaCorners

AreaManager.AreaManager, 41

ReadConfigFile

GeoSams.MainApplication, 71

ReadFields

FishMortBySpecAcc.FishMortBySpecAcc, 53

ReadFile

EditMathSetupFrame.EditMathSetup, 49

ReadGridMgrConfigFile

GeoSams.MainApplication, 72

ReadRecruitConfigFile

GeoSams.MainApplication, 72

ReadSimConfigFile

GeoSams.MainApplication, 72

recNum

shapefile.Writer, 131

record

shapefile.Reader, 91

shapefile.ShapeRecord, 101

shapefile.Writer, 130

ShapeTest, 31

SortIntoColumns, 34

records

shapefile.Reader, 91

RECR

Globals, 23

recrCfgFile

MainInputFrame.MainInput, 83

recrConfigFile

GeoSams.MainApplication, 74

recruitYrStop

GeoSams.MainApplication, 74

recruitYrStrt

GeoSams.MainApplication, 74

recrVar

MainInputFrame.MainInput, 83

recrYrStop

MainInputFrame.MainInput, 83

recrYrStrt

MainInputFrame.MainInput, 83

Region

SortIntoColumns, 34

SortIntoColumns.GeoShape, 57

results

AreaManager.AreaMgrSubFrame, 45

resultsDir

Globals, 24

SortByRegionFrame.SortByRegion, 113

rewind

shapefile, 27

RING

shapefile, 29

ring\_bbox

shapefile, 27

ring\_contains\_point

shapefile, 27

ring\_contains\_ring

shapefile, 27

ring\_sample

shapefile, 27

root

FishMortBySpecAcc.FishMortBySpecAcc, 54

GeoSams.MainApplication, 74

GrowthFrame.Growth, 66

MainInputFrame.MainInput, 83

SortByAreaFrame.SortByArea, 108

SortByRegionFrame.SortByRegion, 114

SpecialAreaFrame.SpecialArea, 118

rows

SortIntoColumns, 34

Run\_Sim

GeoSams.MainApplication, 72

RunSort

SortByAreaFrame.SortByArea, 106

SortByRegionFrame.SortByRegion, 113

runSortButton

SortByAreaFrame.SortByArea, 108

SortByRegionFrame.SortByRegion, 114

SAMS

ShapeTest, 31

ShapeTest.GeoShape, 55

SaveAreaFile

SpecialAreaFrame.SpecialArea, 117

saveAreaFileButton

SpecialAreaFrame.SpecialArea, 118

SaveConfigFiles

GeoSams.MainApplication, 72

saveDataSortButton

SortByAreaFrame.SortByArea, 108

SaveDataSortFile

SortByAreaFrame.SortByArea, 106

SaveField

AreaManager.AreaMgrSubFrame, 44

SaveFMFile

FishMortBySpecAcc.FishMortBySpecAcc, 53

saveFMFileButton

FishMortBySpecAcc.FishMortBySpecAcc, 54

saveGrowthConfigButton

GrowthFrame.Growth, 66

SaveSpecialAreaData

SpecialAreaFrame.SpecialArea, 117

scrollFrame

AreaManager.AreaManager, 42

SortByAreaFrame.SortByArea, 108

SortByRegionFrame.SortByRegion, 114

scrollFrameHeight

Globals, 24

scrollFrameWidth

Globals, 24

sep

SortIntoColumns, 34

Set Fishing Mortality in Special Access Areas, 4

setDredgeDataButton

MainInputFrame.MainInput, 83

SetDredgeFileEnvVar

MainInputFrame.MainInput, 81

SetDredgeFileName

MainInputFrame.MainInput, 81

SetGbShapeBufferFile

MainInputFrame.MainInput, 81

setGbShapeBufferFileButton

MainInputFrame.MainInput, 83

SetGBShapeBufferFileEnvVar

MainInputFrame.MainInput, 81

SetGbShapeFile

MainInputFrame.MainInput, 81

setGbShapeFileButton

MainInputFrame.MainInput, 83

SetGBShapeFileEnvVar

MainInputFrame.MainInput, 81

setHabCamDataButton

MainInputFrame.MainInput, 83

SetHabCamFileEnvVar

MainInputFrame.MainInput, 81

SetHabCamFileName

MainInputFrame.MainInput, 81

SetMaShapeBufferFile

MainInputFrame.MainInput, 81

setMaShapeBufferFileButton

MainInputFrame.MainInput, 83

SetMaShapeBufferFileEnvVar

MainInputFrame.MainInput, 81

SetMaShapeFile

MainInputFrame.MainInput, 81

setMaShapeFileButton

MainInputFrame.MainInput, 83

SetMaShapeFileEnvVar

MainInputFrame.MainInput, 81

sf

ShapeTest, 31

SortIntoColumns, 34

shape

shapefile.Reader, 91

shapefile.ShapeRecord, 101

shapefile.Writer, 130

SortIntoColumns, 34

shapefile, 24

\_\_version\_\_, 28

b, 26

bbox\_contains, 26

bbox\_overlap, 26

failure\_count, 28

FIRST\_RING, 28

INNER\_RING, 28

is\_cw, 26

is\_string, 26

izip, 28

logger, 28

MISSING, 28

MULTIPATCH, 28

MULTIPOINT, 28

MULTIPOINTM, 28

MULTIPOINTZ, 28

NODATA, 28

NULL, 28

organize\_polygon\_rings, 26

OUTER\_RING, 28

PARTTYPE\_LOOKUP, 28

pathlike\_obj, 27

POINT, 29

POINTM, 29

POINTZ, 29

POLYGON, 29

POLYGONM, 29

POLYGONZ, 29

POLYLINE, 29

POLYLINEM, 29

POLYLINEZ, 29

PYTHON3, 29

rewind, 27

RING, 29

ring\_bbox, 27

ring\_contains\_point, 27

ring\_contains\_ring, 27

ring\_sample, 27

SHAPETYPE\_LOOKUP, 29

signed\_area, 27

test, 28

TRIANGLE\_FAN, 29

TRIANGLE\_STRIP, 29

u, 28

VERBOSE, 29

xrange, 29

shapeFile

SortIntoColumns, 34

shapefile.\_Array, 36

\_\_repr\_\_, 36

shapefile.\_Record, 36

\_\_dir\_\_, 38

\_\_field\_positions, 39

\_\_getattr\_\_, 38

\_\_getitem\_\_, 38

\_\_init\_\_, 38

\_\_oid, 39

\_\_repr\_\_, 39

\_\_setattr\_\_, 39

\_\_setitem\_\_, 39

as\_dict, 39

oid, 39

shapefile.Reader, 85

\_\_dbfHdrLength, 93

\_\_dbfHeader, 88

\_\_del\_\_, 88

\_\_enter\_\_, 88

\_\_exit\_\_, 89

\_\_fieldLookup, 93

\_\_fullRecLookup, 93

\_\_fullRecStruct, 93

\_\_geo\_interface\_\_, 89

\_\_getFileObj, 89

\_\_init\_\_, 88

\_\_iter\_\_, 89

\_\_len\_\_, 89

\_\_record, 89

\_\_recordFields, 89

\_\_recordFmt, 89

\_\_recordLength, 93

\_\_restrictIndex, 89

\_\_shape, 90

\_\_shapeIndex, 90

\_\_shpHeader, 90

\_\_shxHeader, 90

\_\_shxOffsets, 90

\_\_str\_\_, 90

\_files\_to\_close, 93

\_offsets, 93

bbox, 93

close, 90

dbf, 93

encoding, 93

encodingErrors, 93

fields, 93

iterRecords, 90

iterShapeRecords, 90

iterShapes, 91

load, 91

load\_dbf, 91

load\_shp, 91

load\_shx, 91

mbox, 93

numRecords, 93

numShapes, 93

record, 91

records, 91

shape, 91

shapeName, 93

shapeRecord, 92

shapeRecords, 92

shapes, 92

shapeType, 93

shapeTypeName, 93

shp, 93

shpLength, 93

shx, 93

zbox, 93

shapefile.Shape, 96

\_\_geo\_interface\_\_, 98

\_\_init\_\_, 98

\_\_oid, 99

\_\_repr\_\_, 98

\_errors, 99

\_from\_geojson, 98

oid, 98

parts, 99

partTypes, 99

points, 99

shapeType, 99

shapeTypeName, 98, 99

shapefile.ShapefileException, 99

shapefile.ShapeRecord, 100

\_\_geo\_interface\_\_, 100

\_\_init\_\_, 100

record, 101

shape, 101

shapefile.ShapeRecords, 101

\_\_geo\_interface\_\_, 101

\_\_repr\_\_, 101

shapefile.Shapes, 102

\_\_geo\_interface\_\_, 102

\_\_repr\_\_, 102

shapefile.Writer, 123

\_\_bbox, 126

\_\_dbfHeader, 126

\_\_dbfRecord, 126

\_\_del\_\_, 126

\_\_enter\_\_, 126

\_\_exit\_\_, 126

\_\_getFileObj, 127

\_\_init\_\_, 126

\_\_len\_\_, 127

\_\_mbox, 127

\_\_shapefileHeader, 127

\_\_shpFileLength, 127

\_\_shpRecord, 127

\_\_shxRecord, 127

\_\_zbox, 127

\_bbox, 131

\_files\_to\_close, 131

\_mbox, 131

\_shapeparts, 127

\_zbox, 131

autoBalance, 131

balance, 127

bbox, 128

close, 128

dbf, 131

deletionFlag, 131

encoding, 131

encodingErrors, 131

field, 128

fields, 131

line, 128

linem, 128

linez, 128

mbox, 128

multipatch, 128

multipoint, 129

multipointm, 129

multipointz, 129

null, 129

point, 129

pointm, 129

pointz, 129

poly, 129

polym, 129

polyz, 130

recNum, 131

record, 130

shape, 130

shapeType, 131

shapeTypeName, 130

shp, 131

shpNum, 131

shx, 131

target, 131

zbox, 130

shapeFileDir

Globals, 24

MainInputFrame.MainInput, 83

shapeGB

ShapeTest, 31

shapeLen

ShapeTest, 31

SortIntoColumns, 34

shapeMA

ShapeTest, 31

shapeName

shapefile.Reader, 93

shapeRecord

shapefile.Reader, 92

shapeRecords

shapefile.Reader, 92

shapes

shapefile.Reader, 92

ShapeTest, 31

SortIntoColumns, 34

ShapeTest, 30

areaKm2, 31

as\_dict, 31

lat, 31

lon, 31

NewSAMS, 31

pointLen, 31

record, 31

SAMS, 31

sf, 31

shapeGB, 31

shapeLen, 31

shapeMA, 31

shapes, 31

X, 31

Y, 31

ShapeTest.GeoShape, 55

\_\_init\_\_, 55

areaKm2, 55

lat, 55

lon, 55

NewSAMS, 55

SAMS, 55

X, 55

Y, 55

ShapeTest.py, 135

shapeType

shapefile.Reader, 93

shapefile.Shape, 99

shapefile.Writer, 131

SHAPETYPE\_LOOKUP

shapefile, 29

shapeTypeName

shapefile.Reader, 93

shapefile.Shape, 98, 99

shapefile.Writer, 130

shapeVal

Widgets.SubFrameInterpFunction, 121

ShowArgs

GeoSams.MainApplication, 72

ShowMessage

Globals, 22

shp

shapefile.Reader, 93

shapefile.Writer, 131

shpLength

shapefile.Reader, 93

shpNum

shapefile.Writer, 131

shx

shapefile.Reader, 93

shapefile.Writer, 131

signed\_area

shapefile, 27

simCfgDir

Globals, 24

simCfgFile

MainInputFrame.MainInput, 83

simConfigFile

GeoSams.MainApplication, 75

simStartDir

MainInputFrame.MainInput, 83

SpecialAreaFrame.SpecialArea, 118

skipStatusMsgs

GeoSams.MainApplication, 75

skipStatusMsgsRB

GeoSams.MainApplication, 75

Sort By Area Frame, 9

Sort By Region Frame, 11

sortAreaFrame

AreaManager.AreaManager, 42

SortByRegionFrame.SortByRegion, 114

SortByAreaFrame, 32

SortByAreaFrame.py, 136

SortByAreaFrame.SortByArea, 102

\_\_init\_\_, 105

AppendYears, 106

areaData, 107

areaFName, 107

areas, 107

browseExportButton, 107

BrowseExportFile, 106

CbUpdateUnits, 106, 107

comboParameter, 107

dataSortFileEntry, 107

dataSortFileLabel, 107

dataSortUnitsEntry, 107

dataSortUnitsLabel, 107

domainName, 107

EnterKeyClicked, 106, 107

ExportAll, 106

exportAllSortButton, 107

exportFileEntry, 107

exportFileLabel, 107

exportFileName, 107

ExportThis, 106

exportThisSortButton, 107

friend, 107

GetDataSortFile, 106

maxYears, 107

numAreas, 107

numAreasEntry, 108

numAreasLabel, 108

numAreasMax, 108

NumAreasUpdate, 106

numCorners, 108

numCornersMax, 108

numYears, 108

on\_visibility, 106, 108

openDataSortButton, 108

outputParmLabel, 108

paramStr, 108

pop\_up, 106

root, 108

RunSort, 106

runSortButton, 108

saveDataSortButton, 108

SaveDataSortFile, 106

scrollFrame, 108

startDir, 108

UpdateUnits, 106

UpdateWidgets, 106

yearStart, 108

yearStop, 108

SortByRegionFrame, 32

SortByRegionFrame.GeoShape, 56

\_\_init\_\_, 56

area, 56

lat, 56

lon, 56

X, 56

Y, 56

Zone, 56

SortByRegionFrame.py, 136

SortByRegionFrame.SortByRegion, 108

\_\_init\_\_, 111

AppendYears, 111

areaFName, 113

areaKm2, 113

browseExportButton, 113

BrowseExportFile, 111

comboParameter, 113

DetermineMgmtAreaIndex, 111

domainName, 113

ExportAll, 111

exportAllSortButton, 113

exportFileEntry, 113

exportFileLabel, 113

exportFileName, 113

ExportThis, 111

exportThisSortButton, 113

firstYrCol, 113

friend, 113

GetZoneData, 112

maxAreas, 113

maxYears, 113

numAreas, 113

numCorners, 113

numYears, 113

on\_visibility, 112, 113

outputParmLabel, 113

paramStr, 113

pop\_up, 112

resultsDir, 113

root, 114

RunSort, 113

runSortButton, 114

scrollFrame, 114

sortAreaFrame, 114

startDir, 114

table, 114

tableCols, 114

tableRows, 114

UpdateWidgets, 113

yearStart, 114

yearStop, 114

zones, 114

SortIntoColumns, 32

as\_dict, 34

columns, 34

dataFile, 34

domain, 34

fileName, 34

index, 34

inputFile, 34

l, 34

M, 34

na\_rep, 34

name, 34

nodes, 34

outfile, 34

pointLen, 34

record, 34

Region, 34

rows, 34

sep, 34

sf, 34

shape, 34

shapeFile, 34

shapeLen, 34

shapes, 34

subDir, 35

X, 35

X\_t, 35

Y, 35

Y\_t, 35

SortIntoColumns.Column, 46

\_\_init\_\_, 46

inBox, 46

name, 46

SortIntoColumns.GeoShape, 57

\_\_init\_\_, 57

lat, 57

lon, 57

Region, 57

X, 57

Y, 57

SortIntoColumns.py, 136

specAccCfgDir

Globals, 24

specAccFile

SpecialAreaFrame.SpecialArea, 118

specAccFileLabel

SpecialAreaFrame.SpecialArea, 118

specAccFileStr

GeoSams.MainApplication, 75

Special Access Area, 13

SpecialAreaFrame, 35

SpecialAreaFrame.py, 137

SpecialAreaFrame.SpecialArea, 114

\_\_init\_\_, 116

areaFName, 118

areaMgr, 118

EnterKeyClicked, 117, 118

friend, 118

GetAreaFile, 117

GetGMgrConfigFName, 117

gmCfgFile, 118

numAreas, 118

numAreasEntry, 118

numAreasLabel, 118

numAreasMax, 118

NumAreasUpdate, 117

numCorners, 118

numCornersMax, 118

on\_visibility, 117, 118

openAreaFileButton, 118

openGmgrConfigButton, 118

pop\_up, 117

root, 118

SaveAreaFile, 117

saveAreaFileButton, 118

SaveSpecialAreaData, 117

simStartDir, 118

specAccFile, 118

specAccFileLabel, 118

startDir, 118

UpdateWidgets, 117

startDayComboDay

MainInputFrame.MainInput, 83

startDayComboMonth

MainInputFrame.MainInput, 83

startDayLabel

MainInputFrame.MainInput, 83

startDir

FishMortBySpecAcc.FishMortBySpecAcc, 54

SortByAreaFrame.SortByArea, 108

SortByRegionFrame.SortByRegion, 114

SpecialAreaFrame.SpecialArea, 118

startincCol

AreaManager.AreaMgrSubFrame, 45

startYr

MainInputFrame.MainInput, 83

stopDayComboDay

MainInputFrame.MainInput, 83

stopDayComboMonth

MainInputFrame.MainInput, 83

stopDayLabel

MainInputFrame.MainInput, 84

stopYr

MainInputFrame.MainInput, 84

style

GeoSams.MainApplication, 75

subDir

SortIntoColumns, 35

surveyDataDir

Globals, 24

surveyStartDir

MainInputFrame.MainInput, 84

table

SortByRegionFrame.SortByRegion, 114

tableCols

SortByRegionFrame.SortByRegion, 114

tableRows

SortByRegionFrame.SortByRegion, 114

target

shapefile.Writer, 131

test

shapefile, 28

ToggleSkipStatusMsgs

GeoSams.MainApplication, 72

towSpdStr

GrowthFrame.Growth, 66

towSpeed

GrowthFrame.Growth, 66

TRIANGLE\_FAN

shapefile, 29

TRIANGLE\_STRIP

shapefile, 29

tsPerYear

GeoSams.MainApplication, 75

MainInputFrame.MainInput, 84

u

shapefile, 28

UpdateEntry

Globals, 22

UpdateUnits

SortByAreaFrame.SortByArea, 106

UpdateValues

GrowthFrame.Growth, 63

UpdateWidgets

AreaManager.AreaManager, 42

FishMortBySpecAcc.FishMortBySpecAcc, 53

GrowthFrame.Growth, 63

SortByAreaFrame.SortByArea, 106

SortByRegionFrame.SortByRegion, 113

SpecialAreaFrame.SpecialArea, 117

useHabCamData

GeoSams.MainApplication, 75

useMatlabRB

EditMathSetupFrame.EditMathSetup, 50

useOctaveRB

EditMathSetupFrame.EditMathSetup, 50

usingMatlab

EditMathSetupFrame.EditMathSetup, 50

VERBOSE

shapefile, 29

viewPort

Widgets.ScrollFrame, 96

Widgets, 35

floatCallback, 35

numbersCallback, 35

Widgets.py, 137

Widgets.ScrollFrame, 94

\_\_init\_\_, 95

canvas, 96

canvas\_window, 96

onEnter, 95, 96

onFrameConfigure, 95, 96

onLeave, 95, 96

onMouseWheel, 95, 96

viewPort, 96

Widgets.SubFrameElement, 119

\_\_init\_\_, 119

myEntry, 120

myLabel, 120

Widgets.SubFrameInterpFunction, 120

\_\_init\_\_, 121

dimVal, 121

funcFrame, 121

myDimRBx, 121

myDimRBy, 121

myDimRBz, 121

myShapeC, 121

myShapeG, 121

myShapeL, 121

myShapeS, 121

preconEntry, 121

preconLabel, 121

shapeVal, 121

Widgets.SubFrameXY, 122

\_\_init\_\_, 122

cornerFrame, 123

latitude, 123

longitude, 123

WriteGridMgrConfig

GeoSams.MainApplication, 72

WriteGrowthConfig

GeoSams.MainApplication, 72

WriteRecruitmentConfig

GeoSams.MainApplication, 72

WriteScallopConfig

GeoSams.MainApplication, 72

WriteSpatialFncsConfig

GeoSams.MainApplication, 73

WriteStartupFile

EditMathSetupFrame.EditMathSetup, 49

X

ShapeTest, 31

ShapeTest.GeoShape, 55

SortByRegionFrame.GeoShape, 56

SortIntoColumns, 35

SortIntoColumns.GeoShape, 57

X\_t

SortIntoColumns, 35

xrange

shapefile, 29

Y

ShapeTest, 31

ShapeTest.GeoShape, 55

SortByRegionFrame.GeoShape, 56

SortIntoColumns, 35

SortIntoColumns.GeoShape, 57

Y\_t

SortIntoColumns, 35

yearEntry

FishMortBySpecAcc.FishMortBySpecAcc, 54

yearStart

AreaManager.AreaMgrSubFrame, 45

GeoSams.MainApplication, 75

SortByAreaFrame.SortByArea, 108

SortByRegionFrame.SortByRegion, 114

yearStop

AreaManager.AreaMgrSubFrame, 45

GeoSams.MainApplication, 75

SortByAreaFrame.SortByArea, 108

SortByRegionFrame.SortByRegion, 114

zbox

shapefile.Reader, 93

shapefile.Writer, 130

Zone

SortByRegionFrame.GeoShape, 56

zones

SortByRegionFrame.SortByRegion, 114