

FUプログラミング説明書（暫定版、July, 2014）

おことわり：本原稿は作成途中であり、一部ソースコードと対応していないところがあるかも知れないので、ソースコードで確認していただきたい。

## 1. はじめに

本ソフトウェアは、量子化学計算プログラム GAMESS [1] を用いた fragment molecular orbital method (FMO) [2] 計算を支援する目的で開発されたプログラムで(使用言語は python)、入力データ作成支援と計算結果の可視化を行う。

本ソフトウェアは FMOutil (使用言語は Fortran) の後継であり、GUI を導入することにより、巨大・複雑タンパク質複合体を容易に扱えるようにすることを目標として、「FMO Strdy Group」 [3] により開発されているものである。本プログラムが、FMO 計算のためだけでなく、タンパク質をはじめとする巨大分子の電子状態計算を行う計算化学者の皆様に少しでも役に立つことを願って公開するものである。研究生生活のお供として各自・各研究グループで育てていただければ筆者の望外の喜びである。

本説明書は、読者が python 言語を習得していることを前提とし、fu の構造と各種クラス・関数の使い方について述べる。fu 本体はかなり大きなプログラムなので、最初からこのコードを読んで解読するのは負担が大きすぎると思われるため、本ソフトウェアで用いている基本的な class の用法を学習するための tutorial を用意している。fu のクラス・関数の使い方を学習するために、各 tutorial プログラムの動作確認から始められることをお勧めする。

プログラムを解読する際、コードが合理的に書かれていないと、その意味を理解しようとして大変な時間を浪費することになる。深遠な理由があつてそのように書かれていると思ってしまう。そういう場合もあるが、そうではなくて技術の稚拙さによることも多い。これはプログラミング知識・技術の優劣を反映するもので、コードを書いた人よりも読者のほうが高度がどうか依存する。十分な経験を持つ人は、初心者、中級者が書く無駄なコードを自分の経験に照らして判断できると思う。コードは、高速なプログラムを書くか、保守・拡張性を重視して読みやすいコードにするかによっても違ってくる。本ソフトウェアは後者の立場で書かれている。

コードはすべて合理的に書くべきであるが、筆者は Fortran のプログラミング経験しか無かったところ、本プログラムを開発するために python の勉強を初めてまだ約 1 年の経験しかない初心者である。従つて、python の特徴を生かしたコードがかけているとは思っていない。随所に Fortran 風の無駄な記述が含まれていて、python 流の合理的コードであるとは言いがたいところが随所にあり、非常に稚拙なプログラムである。python のみならず、筆者は GUI プログラムについても初心者である。これらに習熟している諸氏は合理的・簡潔なコードに書き換えて使っていただきたい。

最後に、本ソフトウェアは、部分的に、CMSI [4] の支援を受けて開発されたことを記し

ておく。

[1] [`http://www.msg.ameslab.gov/games/`](http://www.msg.ameslab.gov/games/)

[2] [`http://staff.aist.go.jp/d.g.fedorov/`](http://staff.aist.go.jp/d.g.fedorov/)

[3] [`http://ma.cms-initiative.jp/community`](http://ma.cms-initiative.jp/community)

Everyone who is interested in FMO is welcome to join the FMO User Group.

[4] [`http://www.cms-initiative.jp/ja`](http://www.cms-initiative.jp/ja)

## 2. 利用許諾条件

本ソフトウェアのソースコードはパブリックドメインとして公開する。Copyrightおよび利用許諾条件はthe BSD 2-Clause Licenseに従う。

## 3. Contributors

本ソフトウェアのContributorsは、各モジュールの先頭部分に記載している。本ソフトウェアを改変して配布される場合、改変したモジュールの先頭部分にあるContributors listに自由に自分の名前を追加していただくとありがたい。

## 4. プログラミング環境

本ソフトウェアはpython, numpy, scipy, wxpython, pyOpenGL, Matplotlibを利用している。

これらのsite-packagesは整合性のあるバージョンを使う必要がある。筆者は2012年の7月からpythonの勉強を始めたが、pythonの最新バージョン3.0を使い始めて、コード開発を行っている途中で、numpy, scipyを使おうとして、これらがpython2.7にしか対応していないことが判明した。site-packagesはpythonの新版の公開より遅れるのが普通であり、自分が開発するプログラムで何が必要かを考え（最初から分からず途中であれを使おうこれを使おうとなる場合が多いので厄介であるが）、適切なバージョンを選ばなければ時間の浪費をすることになる（コード開発が進むほどバージョンを変更することに伴う書き換えに必要な時間が増える）。従って、結果的に“保守的”な環境を選択した。64bit版を避けて32bit版にしたのも同じ理由である。

### 4-1 pythonとsite-packagesのインストール

本ソフトウェアはソースコードのみが配布されるので、使用するにはpythonの開発環境を各自のパソコンに自ら構築しなければならない。筆者の開発環境（Window7）は下記のとおりである。以下、参考のために筆者がダウンロードしたサイトを紹介するが、すべて各自が自分の責任でダウンロード先を選択してほしい。

1) Python version2.7のWindows32bit版

<http://www.python.org/download/> で公開されている Python 2.7.5 Windows Installer (Windows binary -- does not include source) をダウンロードした。

2) numpy-1.6.2

<http://sourceforge.net/projects/numpy/files/NumPy/1.6.2/> で公開されている numpy-1.6.2-win32-superpack-python2.7.exe をダウンロードした。

3) scipy-0.11.0b1

<http://sourceforge.net/projects/scipy/files/scipy/0.11.0b1/> で公開されている scipy-0.11.0b1-win32-superpack-python2.7.exe をダウンロードした。

4) wxPython (wx-2.8-msw-unicode)

<http://www.wxpython.org/download.php> で公開されている wxPython2.8-win32-unicode-py27 をダウンロードした。

5) PyOpenGL-3.0.2

<http://www.lfd.uci.edu/~gohlke/pythonlibs/> で公開されている PyOpenGL-3.0.2.win32-py2.7.exe をダウンロードした。

6) matplotlib-1.2.0

<https://github.com/matplotlib/matplotlib/downloads> で公開されている matplotlib-1.2.0.win32-py2.7.exe をダウンロードした。

7) py2exe (py2exe-0.6.9.win32-py2.7)

[http://sourceforge.jp/projects/sfnet\\_py2exe/downloads/py2exe/0.6.9/py2exe-0.6.9.win32-py2.7.exe/](http://sourceforge.jp/projects/sfnet_py2exe/downloads/py2exe/0.6.9/py2exe-0.6.9.win32-py2.7.exe/) からダウンロードした。

#### 4-2 exe (fumodel.exe と fuplot.exe) の作り方

上記、1) から 7) のソフトウェアのインストール終了後、src ディレクトリにある setup.bat を実行すると、.dist フォルダが作成され、その中に fumodel.exe と fuplot.exe が作成される。

#### 4-3 PYTHONPATH

インストール後の、筆者の Windows7 PC の PYTHONPATH (python shell で import sys; sys.path で表示される) の設定は以下のとおりである。

```
['.',  
'C:¥¥Windows¥¥system32¥¥python27.zip',  
'C:¥¥Python27¥¥DLLs',  
'C:¥¥Python27¥¥lib',
```

```
'C:\Python27\lib\plat-win',  
'C:\Python27\lib\lib-tk',  
'C:\Python27',  
'C:\Python27\lib\site-packages',  
'C:\Python27\lib\site-packages\gtk-2.0',  
'C:\Python27\lib\site-packages\win32',  
'C:\Python27\lib\site-packages\win32\lib',  
'C:\Python27\lib\site-packages\Pythonwin',  
'C:\Python27\lib\site-packages\wx-2.8-msw-unicode']
```

#### 4-4 その他

##### 1) 統合プログラム開発環境ソフトウェア

筆者は、Eclipse に PyDev を付加した統合プログラム開発環境を使用している。

Eclipse は、

[http://sourceforge.jp/projects/sfnet\\_eclipse.mirror/downloads/eclipse-SDK-3.7.2-win32.zip](http://sourceforge.jp/projects/sfnet_eclipse.mirror/downloads/eclipse-SDK-3.7.2-win32.zip) からダウンロードした。

PyDev のインストールは、Eclipse の Help メニューの "Install New Software..." で行う。install の仕方は、

<http://www.brainchild.co.jp/blog/develop/2010/08/python-eclipse1.html> などのサイトが参考になる。

##### 2) python の学習

<http://www.python-izm.com/contents/introduction/install.shtml> には、python の勉強に役立つ情報が掲載されている。

#### 5. fu の構造とモジュール

fu は、分子構造モデリングから FMO 計算のための入力データの作成までを行う fumodel と、FMO 計算の結果をグラフに描画する fuplot の 2 つからなる。

##### 5-1 プログラムの構造

fumodel はかなり大きなプログラムで、図 1 に示す構造を持つ。

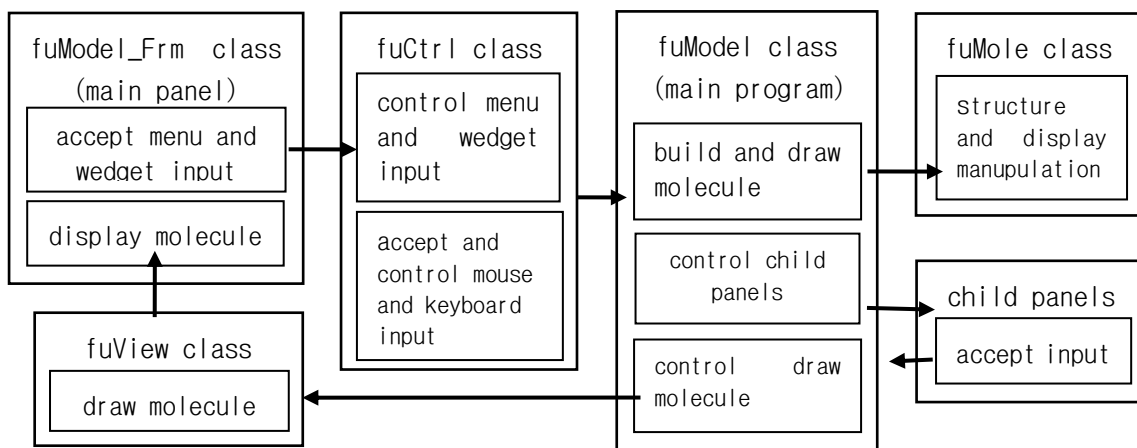


Fig.1 Structure of fumodel program

一方、fuplot は小さなプログラムで、図 2 に示す単純な構造を持つ。ここの分子モデル表示を行う部分は、fumodel プログラムから一部の機能（分子構造の改変に関する機能）を外したサブセットである。

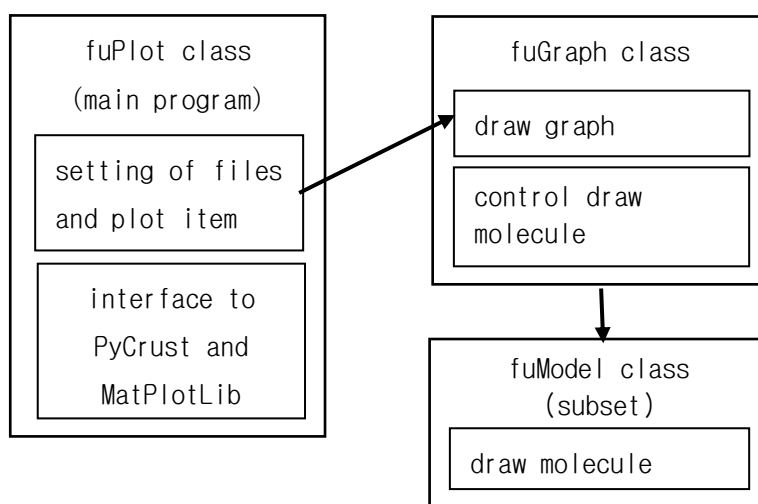


Fig.2 Structure of fuplot program

## 5-2 fu の modules

fu ソフトウェアは次の module からなる。

```
--- modules for fumodel
fumodel.py ... fumodel メインプログラムを含むモジュール。
fuctrl.py ... メニューとマウス・キーボード入力受付と制御ルーチンを含むモジュール
fumole.py ... 分子構造データの保持と構造改変に関する method を含むモジュール
fubuild.py ... 入力受けて分子の表示と構造改変を実行する method を含むモジュール
fupanel.py ... サブウィンドウを集めたモジュール
fuview.py ... 分子モデル描画ルーチンのモジュール
fulib.py ... global function を含むモジュール
fuconst.py ... global constant を含むモジュール
--- modules for fuplot
fuplot.py ... fuplot メインプログラムを含むモジュール
fugrapf.py ... グラフ描画関連のルーチンを含むモジュール
---
setupmdl.py ... py2exe により fumodel.exe を作成するときに用いる
setupplt.py ... py2exe により fuplot.exe を作成するときに用いる
(MS Windows では、これらの exe を作成するには setup.bat を実行すればよい)
```

上記の外、tutorial ディレクトリには、fu では使わないが、fu の class の使い方を勉強するためのいくつかの例が用意されている。

## 5-3 Class and instance of fu program

### 1) Class and instance of fumolde program

```
fumodel (an instance of fuModel class in fumodel.py)
  |-- mollst[mol] attribute (mol: an instance of fuMole class
  |                           in fumole.py)
  |-- mol[atom] attribute (atom: an instance of Atom class
  |                           in fumole.py)
  |-- ctrlflag (an instance of CtrlFlag class in fuctrl.py)
  |-- frame (an instance of fuModel_Frm class in fumodel.py)
  |-- view (an instance of fuView class in fuview.py)
  |-- ctrl (an instance of fuCtrl class in fuctrl.py)
```

```

        |--- menu (an instance of MenuCtrl class in
        |           fuctrl.py)
        |--- input (an instance of InputCtrl class in
        |           fuctrl.py)
|--- pycrust (an instance of wx.py.crust.CrustFrame)
|--- gmsinputwin (an instance of GamessInput_Frm class in
|           fupanel.py)
|--- treewin (an instance of TreeSelector_Frm class in
|           fupanel.py)
|--- namewin (an instance of NameSelector_Frm class in
|           fupanel.py)

```

## 2) Class and instance of fuplot program

```

fuplot (instance of fuPlot class in fuplot.py)
|--- fmodatadic[name] attribute (name: an instance of
|           FMOProperty class in fuplot.py)
|--- graphname[name] attribute (name: an instance of fuGraph
|           class in fuplotpy)
|--- graph (an instance of BarGraph or TileGraph class
|           in fugraph.py)
|--- molview (an instance of fuModel class in fumodel.py)
|--- mpl (an instance of Matplotlib_Frm class in fugraph.py)
|--- figure (an instance of matplotlib.figure)
|--- canvas (an instance of FigureCanvasWxAgg)
|--- pycrust (an instance of wx.py.crust.CrustFrame)

```

## 6. Class and attribute/method of fu

### 6-1 fuModel class (fumodel.py)...main program of fumodel

#### 1) attributes

```

curmol=-1 # the number of current fuMole instance in mollst[]
          # i.e. wrk=mollst[curmol]
mollst=[] # list to store fuMole instance [0,1,...]
molnam=[] # name of molecule [0,1,...]
files=[] # input file name of molecule [0,1,...]
wrkmolnam='' # name of current fuMole instance

```

```

mhtdatadic={} # mht data dic ['name0':[mht data...], 'name1':[mht
                    data], ..}
autoaddhydrogen=True # try to add hydrogen in read PDB data
2) methods
AddBondUseBondLength(): add bond data using bond length
AddBondUseFrameData(): add bond data using molecular frame data
                        (.mht)
AddChainName(): add/change peptide chain name
AddGroup1H(): add one hydrogen to selected atoms
AddGroup2H(): add two hydrogens to selected atoms
AddGroup3H(): add three hydrogens to selected atoms
AddHydrogenToAAResidue(): add hydrogens to selected amino acid
                        residues
AddHydrogenToCutAA(): 断片化したペプチド鎖のCとNに水素原子を付加する
AddHydrogenToWater(): add hydrogens to waters
AddHydrogenUseBondLength(): 結合距離に基づいて水素原子を付加する
AddHydrogenUseFrameData(): フレームデータ(.mht)に基づいて水素原子を
                        付加する
AddItemCBoxMol(molnam): 読み込んだ分子構造データを check box に追加
AssignAAAtomCharge(): assign charge to amino acid residue atoms
AssignIonChg(chg): assign charge to ion
AssignLayer(layer): assign selected atoms to specified layer
                    layer(int): layer number
AssignLayerUndo(): Undo layer assignment (under construction,
                    15May2013)
AtomAngle(atm0, atm1, atm2): get angle of atoms
    atm0(int): sequence number of atom
    atm1(int): sequence number of atom
    atm2(int): sequence number of atom
    ret=angle(float): angle between atm0-atm1 and atm2-atm1
AutoAddHydrogen(on): auto addition of hydrogens to polypeptide
                    and waters in reading PDB file
    on(True/False): on/off flag
BuildMol(filnam, mol): make fuMole instance from data in PDB file
    filename(string): file name
    mol(fuMole instance): None except merge molecule

```



ChangeAtomColor(item,col): change atom color  
     item(string): key word, 'by element','by chain',  
                   or 'color palet'  
     col(list): color  
 ChangeBondKind(atmpairlst,bndknd): change bond kind. under  
                   constraction(15 Mat 2013)  
 ChangeStickBold(bold): change thickness of bond line  
     bold(int): thickness, default=2  
 CheckShortContact(mol1,mol2): check short contact  
     ret=nsht(int): number of short contact  
         ,rmin(float): shortest distance  
 ClearAllLayer(): cancel all layer assignment  
 ClearBDA(): remove all BDA data  
 ClearClipboard():empty clipboard  
 ClearLayer(layer):cancel layer assginment in specified layer  
     layer(int): layer number  
 ConsoleMessage(mess):print message on PyCrust console  
 CopyMolBitmapToClipboard():copy screen image to clipboard in  
                   bipmap data  
 CopyMolToClipboard(): copy selected atom data to clipboard  
 CountEnvSelAtm():count number of environment atoms  
     ret=nenv(int): number of environment atoms  
         ,lst(list): list of sequence numbers o environment atoms  
 CreateFrame(parent,fumode,size): create fmomodel frame  
 DeleteBonds(kind):delete bond kind data  
 DeleteHydrogens():delete selected hydorogen atoms  
 DeleteNonBonded():delete non-bonded atoms  
 DeleteSelected():delete selected atoms  
 DeleteTerAtoms():delete 'TER' atom (see sec7, 1))  
 DeleteVdwBond():delete vdw contact data (trial base)  
 DeleteWater(): delete selected waters  
 DispAtomAngle(atm0,atm1,atm2):display angle on statusbar  
     atom0(int): atom number  
     atom1(int): atom number  
     atom2(int): atom number  
 DrawBDAPoint(on): draw BDA points

on(True/False): on/off flag  
 DrawChainKite(on): in preparation (15 May 2013)  
 DrawChainTube(on): draw cube peptide chain  
     on(True/False): on/off flag  
 DrawDistance(on): draw interatomic distance  
     on(True/False): on/off flag  
 DrawFormalCharge(on): draw formal charge  
     on(True/False): on/off flag  
 DrawLabelAtm(on,case): draw atom name  
     on(True/False): on/off flag  
     case(int): 0 name only, 1 name and atom number  
 DrawLabelElm(on,case): 元素名を描く  
     case(int): 0 name only, 1 name and sequence number of atom  
     on(True/False): on/off flag  
 DrawLabelFrg(on): draw fragment name  
     on(True/False): on/off flag  
 DrawLabelRes(on,case): draw residue name  
     case(int): 0 name only, 1 name and residue number  
     on(True/False): on/off flag  
 DrawLayer(on): draw molde by layer color  
     on(Tre/False): on/off flag  
 DrawMol(updated): draw molecule  
     update(True/False): flag to force to make makedisplay list  
 DrawPlotData(on,save,item,pltcol,ifrg): (fuplot プログラム) で  
     plot data を描く on/off flag  
 DrawVdwBond(on): vdW 結合線を描く on/off flag (trial base)  
 FindBondSeqNmb(atm1,atm2): 結合している原子の番号を見つける  
     ret=nmb  
 FindFragNmb(frgrnam): fragment の番号を見つける  
     ret=ifrg  
 FindGrpNmb(ia,grplst): グループの番号を見つける (未実装)  
     ret=igr  
 FindMinMaxXYZ(mol): Decart だ表の最大値と最小値を見つける  
     ret=xmin,xmax,ymin,ymax,zmin,zmax  
 FindRadiusAtoms(radius): 選択状態にある原子から半径 radius 内にある原  
     子を見つける

```

ret=nmb,lst
FindRadiusResidue(radius):選択状態にある原子から半径 radius 内にあ
る残基を見つける
ret=nmb,lst
FitToScreen(draw):分子モデルをスクリーンサイズに合わせて再描画する
FogEnable(on):Fog の on/off flag
FragmentAARes(case):アミノ酸残基を fragment に分割する
FragmentNonAARes():非アミノ酸残基を fragment に分割する
GetFragName(ifrg):ifrg 番目のフラグメント名を取得する
ret=nam
GetMaxAtmNmb():原子の番号の最大値を取得する
ret=atmnmb
GetMaxResNmb():残基の番号の最大値を取得する
ret=resnmb
GetMouSelMode():マウスの選択モードを取得する
GetMouseMoveVector(dif):マウスの移動量を取得する
ret=mov
GetMouseRotMatrix(dif):マウスの移動量に応じた回転行列を取得する
ret=rotmat
GetPntAtm(pos,raspos):マウスで click された原子の番号を取得する
ret=nmb
GetRaspos(ith,raspos):ith 番目の原子の raster position を取得する
ret=inix,iniy
HideEnvironment(on):environment 原子の表示 on/off flag
HideHydrogen(on):水素原子の表示 on/off flag
HideSelected(on):選択状態にある原子の表示 on/off flag
HideWater(on):水分子の表示 on/off flag
InputAtomName(a):入力した名前を原子 a に付ける
InputBDABAA(bda,baa):BDA,BAA を入力する
ListAtomName():原子名の list を作る
ret=lst
ListChainName():ペプチド鎖名の list を作る
ret=lst
ListDrawAtom():描画する原子の list を作る
ret=nmb,lst
ListElement():元素名の list を作る

```

```

    ret=lst
listFragmentName(): fragemnt 名の list を作る
    ret=lst
ListGroupName(): group 名の list を作る (未実装)
    ret=lst
ListResidueName(): 残基名の list を作る
    ret=lst
ListSelectedAtom(): 選択状態にある原子の list を作る
    ret=nsel, lst
ListTargetAtom(): 操作対象の原子の list を作る
    ret=lst
MakeAtomLabel(atm, nameonly): 原子の label を作る
    ret=lbl
MakeBondedAtomGroupList(lst): 結合している原子グループの list を作る
    ret=lst
MakeDrawBDADData(): BDA の描画データを作る
    ret=drwdat
MakeDrawDistanceData(): 距離描画のデータをつくる
    ret=disdat
MakeEnvByList(lst): lst により environment 原子を設定する
MakeEnvGrpByRadius(selobj, radius): 選択状態の原子から radius 以内
    にある対象物を environment に設定する
    ret=nmbr, lst
MakeNewMol(case): 新分子を作る (原子がゼロの FuMole instance を作る)
MakeNonAAResLst(): 非アミノ酸残基の list を作る
    ret=reslst
MakeVdwContact(): Vdw contact 情報を作る
ManualBDASetting(on): BDA を手動で設定するモードの on/off flag
MergeFragments(): fragment を merge する
MergeMol(mol): カレント分子に mol 分子を merge する
MergeToCurrent(filename): current 分子に PDB file の分子を merge
    する
MessageStatus(mess, loc, col): statusbar に message を出力する
MessAtomAngle(atm0, atm1, atm2): statusbar に原子間角を出力する
MessAtomDistance(atm0, atm1): statusbar に原子間距離を出力する
MessAtomLabel(atm, nameonly): statusbar に原子名を出力する

```

NewMolecule():新しい分子を作る  
 OpenAssignLayerPanel():layer assign panel を open する  
 OpenAtmChgPanel():原子電荷入力 panel を open する  
 OpenControlPanel():control panel を open する  
 OpenGamessPanel():GAMESS input assistant panel を open する  
 OpenNameSelector(winpos):Name/number selector panel を open する  
 OpenPyCrust():PyCrust console を open する  
 OpenTreeSelector(winpos):Tree selector panel を open する  
 OrgOrient(pdborg,pdbnew):分子を入力時の配向にする  
     ret=err  
 PasteMolFromClipboard():clipboard の分子データを paste する  
 PrintMessage():Message を print する  
 Quit():program を終了する  
 ReadFiles(filename):file を読む  
 Redraw():画面を再描画する  
 RemoveEnvGroup():environment グループを解除する  
 RemoveMol(allmol):カレント分子または全分子を消去する  
 RenameItemCBoxMol(itmnm, newname):ComboBox の分子名を rename する  
 RenameMolecule(filename):仮名の分子を file 名の名前に変更する  
 ResetItemdicCheck(choosen, check):checkable menu item の check  
 を  
     reset する  
 ResetPosAtm():マウスで前回 point された原子のデータを reset する  
 ResetShowAtom(on):原子の表示 on/off 状態を reset する  
 RotateSelected(dif):選択原子集団を回転する  
 SaveAtomColor(save):原子色データを backup する  
 SaveFragmentAs(filename):fragment 情報データを file に保存する  
 SaveLayerData(save):layer データを backup する  
 SaveRasPosZ(value):raster position データを backup する  
 SaveShwAtm(value):原子表示データを backup する  
 SelectAABackbone():ポリペプチドの主鎖を選択状態にする  
 SelectAAResidue():アミノ酸残基を選択状態にする  
 SelectAASideChain():ポリペプチドの側鎖を選択状態にする  
 SelectAll(selflg):全原子を選択/非選択状態にする  
 SelectAllShowAtom():show atom flag が on の全原子を選択状態にする  
 SelectAtmNam(atmnam, selflg):原子名 atmnam の原子を選択/非選択にする

SelectAtomByAtmNmb (lst, selflg) : 原子番号 atmnm の原子を選択状態にする

SelectAtomByList (lst, selflg) : list に記載した原子を選択状態にする

SelectAtomByNamNmb (atmnam, atmnm, selflg) : 原子の名前と番号を指定して選択状態にする

SelectAtomBySeqNmb (lst, selflg) : 通し番号を指定して原子を選択状態にする

SelectByCircle (newx, newy, centeratm, raspos) : マウスで drug した球内の原子を選択状態にする

SelectByClick (pntatmhis, selflg) : マウスで click した原子を選択状態にする

SelectByRadius (selobj, radius, flgval) : 選択状態にある原子から半径 radius 以内にある原子を選択状態にする

ret=nselect

SelectByRectangle (inix, iniy, newx, newy, raspos) : マウスで drug した矩形内の原子を選択状態にする

SelectChainByAtmSeqNmb (a, selflg) : 指定した通し番号を持つ原子が属する chain を選択状態にする

SelectChainByList (lst, selflg) : list に記載した chain を選択状態にする

ret=nselect

SelectChainNam (chanam, selflg) : 名前で chain を選択する

SelectComplement () : 選択/非選択を反転させる

SelectElmNam (elmnam, selflg) : 元素名で原子を選択する

SelectEnv (selflg) : environment にアサインされた原子を選択状態にする

ret=nselect

SelectFragByAtmSeqNmb (a, selflg) : 指定した通し番号の原子が属する fragment を選択状態にする

SelectFragNam (frgnam, selflg) : fragment 名で fragment を選択状態にする

SelectHydrogen () : 水素原子を選択状態にする

SelectNonAARes (res, nmb) : res 名と res 番号を持つ非アミノ酸残基を選択状態にする

SelectNonAAResidue () : 非アミノ酸残基を選択状態にする

SelectNonBonded () : 非結合原子を選択状態にする

SelectRes (resnamlst, resnmblst, selflg) : 名前と番号の list で指定した

残基を選択状態にする

SelectResByAtmSeqNmb(a, selflg) : 指定した通し番号を持つ原子が属する残基を選択状態にする

SelectResByNmb(resnmb1st, selflg) : 残基の通し番号 list でしていた残基を選択状態にする

SelectResidueByList(1st, selflg) : list で指定した残基を選択状態にする

ret=nsel

SelectResNam(resnam, selflg) : rewsnam 名の残基を選択状態にする

SelectWater() : 水分子を選択状態にする

SetBDAInAAResidue(target) : アミノ酸残基に BDA を設定する

SetBDAWrkMol(resnam, bdalst) : current 分子に BDA を設定する

SetBondKind(atm1, atm2, bndknd) : 未実装

ret=nset

SetChainColor() : chain のカラーを設定する

ret=ret

SetChosenColor(col) : 選択した色をセットする

ret=nmb

SetDeselectAll() : 全原子の選択状態を解除する

ret=nmb

SetDrawLabelFragmentName() : fragment 名描画データをセットする

SetDrawVdwBond(on) : vdw 結合描画データをセットする

SetElementColor() : 元素カラーをセットする

ret=nmb

SetEnvGrpAtom(ith, on) : environment 原子をセットする

SetFragmentFMOInp(frgrnam, indat, bdabaa) : FMO input データの fragment 名、fragment 原子グループ、BDA データを分子データにセットする

SetGroupEnvFlg(env1st) : environment 原子 flag をセットする

SetLayerColor() : layer のカラーをセットする

SetResidueColor() : 残基カラーをセットする

ret=nmb

SetSection(rot) : 断面を切る配向をセットする

SetSelectAll(selflg) : 全原子を選択/非選択状態にセットする

SetSelectAllAtom(mol, selflg) : 分子 mol の全原子を選択/非選択状態にセットする

SetSelectedAtom(ith, selflg) : ith 番目の原子を選択/非選択状態にセット

する

```
SetSelectRes(resnam, resnmb, selflg) :
    ret=nmb

SetTextFont() : text 用の font を設定する
SetTurnLst() : 分子の順番 list をセットする
SetUpDraw() : 分子の描画のためのデータを設定する
SetWrkMol(im) : カレント分子を設定する
ShowAABackboneOnly(on) : ポリペプチドの主鎖のみ表示 on/off flag
ShowAASideChainOnly(on) : ポリペプチドの側鎖のみ表示 on/off flag
ShowAllAtom() : 全原子の表示 flag を on にする
ShowDrawModel(model) : 分子モデルを変更する
ShowSelectedOnly(on) : 選択状態の原子のみ表示 on/off flag
SwitchMol(name) : current 分子を切り替える
TranslateSelected(dif) : 選択状態の原子 (集団) を併進させる
UpdateMol(mol) : mol 分子の update flag を on にする (描画のため)
WriteFiles(filename, save) : 分子データを file に書き出す
@static method
SplitAtTER(molnam, pdbmol) : TER で分子を分割する
    ret=namdic, datdic, delcondic
SplitWater(molnam, pdbmol) : 分子データに含まれる水分子を切り分ける
    ret=namdic, datdic
```

6-2 fuMole class (fumole.py)・・・分子構造・分子モデル描画のデータ保持と  
改変

1) attributes

```
molname='' # name of molecule, made from input file name
inpfile='' # input file name
outfile='' # save file name
inpform='' # pdb, xyz, fmoinp, gmsinp, zmt (only pdb is supported)
remark='' # comment
mol=[] # list of atom instance
bdadic={} # BDA data for fragments
```

2) methods

```
AddAtomAt(at, elm, coord, atmdatdic) : 1 原子を付加する
    at: int, at 番目の原子の次に加える
    elm:string*3, element
```



```

    coord: list[x,y,z], coordinate
    atmdatdic: dictionary, atom data, see Atom.SetAtomData
AddBDABond(ia,ib) : BDA を付加する
    ia: int, sequence number of bda atom
    ib: int, sequence number of baa atom
AddBond(a,b,multi) : 結合データを付加する
    a: int, sequence number of atom in the bond
    b: int, sequence number of partner atom
    multi: int, bond kind(1:'single',2:'double',3:'triple',
        4:'aromatic',5:'HB',6:'CH/pi',7:'vdw')
AddBondInAARes(resnam,resdat) : アミノ酸残基に結合データを付加する
    resnam: string*3, residue name
    resdat: list, ExtractAARes methodで作る residue データ
AddBondUseBL(lst) : 結合距離に基づいて結合データを付加する。
    lst: list, 対象原子番号のリスト。[]の場合は全原子を対象とする。
AddBondUseMht(lst,mhtdatadic) : 分子フレームデータ(mht)を用いて結合
データを付加する
    lst: list, 操作対象原子のリスト。[]の場合は全原子を対象とする
    mhtdatadic: dictionary of mht data
AddGroup1Hydrogen(lst) : 原子に1個の水素原子を付加する
    lst: list, 操作対象原子のリスト。[]の場合は全原子を対象とする
AddGroup2Hydrogen(lst) : 原子に2個の水素原子を付加する
    lst: list, 操作対象原子のリスト。[]の場合は全原子を対象とする
AddGroup3Hydrogen(lst) : 原子に3個の水素原子を付加する
    lst: list, 操作対象原子のリスト。[]の場合は全原子を対象とする
AddHydrogen(at,nh,coord,hnam) : 原子に水素原子を付加する
    at: int, sequence number of atom to be attached hydrogen(s)
    nh: int, number of hydrogens
    coord: list, [[x,y,z],[x,y,z],...]
    hnam: list, ['atom name',...]
AddHydrogenToAARes(resnam,resdat,ic) : アミノ酸残基に水素原子を付加す
る
    resnam: string*3, residue name
    resdat: list, ExtractAARes methodで作る residue データ
    ic: int, ic >=0 the second and later residue, =-1 the first
        residue

```

```

ret=nh
AddHydrogenToMol(at,hname,htype,bndlst,rhx):分子に水素原子を付加する
    at: int, sequence number of atom to be attached hydrogen(s)
    hname: string*4, atom name of the hydrogen
    htype: string*3, type of add hydrogen, See Sec.6,2)
    bndlst: list, reference atom list, See Sec.6,2)
    rhx: float, bond length
ret=nh
AddHydrogenToNterm(lst): アミノ酸残基の N 末に水素原子を付加する
    lst: list, 対象原子の番号のリスト。[]の場合は全原子を対象
AddHydrogenToPeptideNC(lst): peptide の N と C 末に水素原子を付加する
    lst: list, 対象原子の番号のリスト。[]の場合は全原子を対象
AddHydrogenToProtein(lst): polypeptide に水素原子を付加する
    lst: list, 対象原子の番号のリスト。[]の場合は全原子を対象
AddHydrogenToWaterMol(lst):水分子に水素原子を付加する
    lst: list, 対象原子の番号のリスト。[]の場合は全原子を対象
AddHydrogenUseBL(lst):結合距離に基づいて水素原子を付加する
    lst: list, 対象原子の番号のリスト。[]の場合は全原子を対象
AddHydrogenUseMht(lst,mhtdatadic):分子フレームデータ(mht)を用いて水素原子を付加する
    lst: list, 対象原子の番号のリスト。[]の場合は全原子を対象
    mhtdatadic: dictionary of mht data
AssignAAResAtmChg(): アミノ酸残基の原子に電荷を割り当てる
    ret=err
CenterOfMass(lst):分子の重心と principal moment inertia vector を返す
    lst: list, 対象原子の番号のリスト。[]の場合は全原子を対象
    ret=com,pmi
ChangeAtomName(a,atmnam,atmnm):原子名を変更する
    a: int, sequence number of atom
    atmnam: string*4, atom name
    atmnm: int, atom number
CheckBDADup(ia,ib): BDA の重複をチェックする
    ia: int, sequence number of BDA
    ib: int, sequence number of BAA

```

```

    ret=dup True/False
ClearBDABAA(1st):BDA/BAA データを削除する。
    1st: list, 対象原子の番号のリスト。[]の場合は全原子を対象
CopyMolInstance(): current fuMole instance のコピーを作る
    ret=cpy
CountAARes(): アミノ酸残基を数え上げる
    ret=nres
CountAtomsInRes(resnam,resnmb):アミノ酸残基の原子数を数える
    resnam: string*3, residue name
    resnmb: int, residue number
    ret=natm
CountHydrogen(1st): 水素原子の数を数える
    1st: list, 対象原子の番号のリスト。[]の場合は全原子を対象
    ret=nh
CountHydrogenOfAtom(a):原子についている水素原子の数を返す
    a: int, sequence number of atom
    ret=nh
CountNonAARes():非網にアミノ酸残基の数を返す
    ret=nres
CountResH(resnam,resnmb): アミノ酸残基の水素原子の数を返す
    ret=nh
CountWater(1st):水分子の数を返す
    1st: list, 対象原子の番号のリスト。[]の場合は全原子を対象
    ret=nw
CreateBDA(ia,ib): BDA を設定する
    ia: int, sequence number of BDA atom
    ib: int, sequence number of BAA atom
CreateFrgConDat(): fragmnet 分割のための結合データを作る。See Sec.6,
1)
DelAllKindBonds(1st):全種類の結合データを消去する
DelAtom(ia):原子を削除する
    ia: int, sequence number of atom
DelBDABond(ia,ib):BDA-BAA データを削除する
    ia: int, sequence number of BDA atom
    ib: int, sequence number of BDA atom
DelBond(a,b):結合データを削除する

```

```

        a: int, sequence number of atom
        b: int, sequence number of atom
@staticmethod
DelFrgBDABond(bdalst,ssblst,con): static method
    ret=con
DelHydrogen(lst):水素原子を消去する
    lst: list, 対象原子の番号のリスト。[]の場合は全原子を対象
DelNonBonded(lst):非結合原子を削除する
    lst: list, 対象原子の番号のリスト。[]の場合は全原子を対象
DelVdwBond(lst):vdW 結合データを消去する
    lst(list): 対象原子の番号のリスト。[]の場合は全原子を対象
DelWater(lst):水分子を消去する
    lst: list, 対象原子の番号のリスト。[]の場合は全原子を対象
ExtractAARes(resnam,resnmb):アミノ酸残基データを抽出する
    resnam(string*3): residue name
    resnmb(int): residue number
    ret=nresatm,resdat,resatmdic
FindAddHTypeBL(a):結合距離に基づいて水素原子の付加型 htype を見つける
    a: int, sequence number of atom to be attached hydrogen
    ret=nh,htype,bndlst,rhx
FindAddHTypeMht(a,mhtdat):分子フレームデータに基づいて水素原子の付加
型(addhtype)を見つける
    a: int, sequence number of atom to which hydrogen(s) are
added
    mhtdat: string*3, mht name
    ret=nh,htype,bndlst,rhx
FindAtmNamInRes(atmnam,ist,resnam,resnmb):
    ret=ia
FindCalphaInRes(resnam,resnmb):
    ret=ic
FindCovalentBondedAtom(elmlst,coordlst):
    ret=bndlst
FindItemNmb(lst,b):
    ret=found
FindMaxAtmNmb():
    ret=maxatmnmb

```

```

FindMaxResNmb () :
    ret=maxresnmb
FindNextAtom (ia, atmnam) :
    ret=nmb
FindNmbInGrpLst (ia, grplst) :
    ret=nmb
FindPrevAtom (ia, atmnam) :
    ret=nmb
FindResAtmSeqNmb (resdat, conatm) :
    ret=nmb
FindSSBond (idx) :
    ret=sslst
FrgAARes (lst, case) :
FrgMergeGly () :
    ret=ndel
FrgMergeTwoRes () :
    ret=ndel
FrgNonAARes (lst) :
GetBDADicValue (keywd) :
    ret=val
GetCCAddAtmType1A1 (atmlst, r) :
    ret=nh, coord
GetCCAddAtmType1A2 (atmlst, r) :
    ret=nh, coord
GetCCAddAtmType1A3 (atmlst, r, bang, trans) :
    ret=nh, coord
GetCCAddAtmType2A1 (atmlst, r) :
    ret=nh, coord
GetCCAddAtmType2A2 (atmlst, r) :
    ret=nh, coord
GetCCAddAtmType3A1 (atmlst, r) :
    ret=nh, coord
GetCCOfWaterHydrogen (cow) :
    ret=chw
GetNumberOfBDA () :
GetNumberOfFrgConDat () :

```

```

InterAtomDistance(a,b):
    ret=r
IsBAAAtom(ia):
    ret=ret
IsCaAtCterminal(ia):
    ret=ret
MakeAAResDic():
    ret=resdic
MakeBDALst(idx):
    ret=bdalst
MakeBondedAtomGroupList(condat):
    ret=grplst
MakeChainDic():
    ret=chaindic
MakeDrawAtomData(lst):
    ret=drwdat
MakeDrawBondData(lst):
    ret=drwdat
MakeDrawChainTubeData(lst):
    ret=drwdat
MakeElmDic():
    ret=elmdic
MakeFrgAtmLst(grplst,idx):
    ret=resnam,frglst
MakeFrgConDat(idx):
    ret=con
MakeNonAAResDic():
    ret=resdic
MakeSSBond(lst):
    ret=nss
MergeFrg(lst):
Message(mess,dev,color):
RemoveBDA(ia,ib):
ReNumberAtmNmb():
ReNumberConDat():
ReNumberIndexForFrg():

```

```

        ret=idx
ResetBDADic():
SetBDAAInAARes(lst):
SetBDAAInNonAARes(lst):
    ret=nbda
SetBndMulti(atom1,atom2,bndmulti):
SetDefaultAtmRad(lst):
SetDefaultColor(lst):
SetDefaultVdwRad(lst):
SetFMOXYZAtoms(inpfile):
SetFragmentName():
SetFragmentUseFMOIndat(frgham,indat,bdabaa):
SetPDBAtoms(pdbmol):
SetToBDADic(ia,ib):
WriteFrgDat(filfrg):
WritePDBMol(filename,parnam,parfilnam,con):

@static method
AtmNamElm(atmnam): static method
    ret=elm
BondAtmGrp(condat,ssblst): static method
    ret=grplst
GetOrgSeqNmb(ith,idx): static method
    ret=nmb
MakeFrgTable(frglst): static method
    ret=frgtbl
MakeMolName(filnam): static method
    ret=name
ReadFrgDat(filfrg): static method
    ret=resnam,bdalst
ReadMhtFile(filcon): static method
    ret=resnam,condat
ReadPDBMol(filpdb): static method
    ret=pdbmol
ReadPDBRem(filpdb,key): static method
    ret=item

```

### 6-3 Atom class (fumole.py)・・・原子の構造・描画データの保持

#### 1) attributes

```
seqnmb=-1 # seq number of atoms 0,1,...,natom-1
cc=[] # cartesian coordinate [x,y,z] in Angstrom
conect=[] # connect data
atmnam='' # atom name
atmnmb=-1 # atom number
resnam='' # residue name
resnmb=-1 # residue number
chainnam='' # chain name
altloc=' '
elm='' # element name
focc=0 # occupancy
bfc=0 # thermal factor
charge=0 # atom charge
# additional to pdb data
bndmulti=[] # bond kind.
extrabnd=[] # extra bonds, H-bond,vdW,CH/pi,...
# draw parameters
color=fuconst.ElmCol['ZZ'] # atom color. default:unknown elm
show=True # show flag
select=False # select flag
model=0 # draw model, 0:line model
atmrad=1.0 # scale factor of atom radius for ball and stick model
vdwrad=1.0 # scale factor of van der Waals radius
thick=2 # bond thicknes
# group data
grpnam='' # group name
grpchg=0 # group charge
envflag=False # environment (special group) flag
parnam='' # name of parent molecule
# fragment data
frgnam='' # fragment name, three characters+sequence number
frgchg=0 # fragment atom formal charge used to calculate fragment
charge.
```



```

frgbaa=-1 # atom seq numbe of baa. atom with non zero frgbaa is
           a bda atom.
layer=1 # FMO layer. 1:1st, n: n-th layer and 11:MM in IMOMM,
         12:EFPP
frgcondat=[]

```

## 2) methods

```

GetAtmDataDic():Atomの attributes を dictionary 型で取得する
    ret=atmdatdic
GetDrwParamDic():Atomの描画関連の attributes を dictionary 型で取得
する
    ret=paramdic
GetFrgDataDic():Atomの fragment の attributes を dictionary 型で取
得する
    ret=frgdatdic
GetGrpDatDic():Atomの group 関連の attributes を dictionary 型で取
得する
    ret=grpdatdic
GetResDatDic():Atomの residue 関連の attributes を dictionary 型で
取得する
    ret=resdatdic
SetAtomData(atmdatdic):dictionary 型のデータで Atom の attribute を
設定する
SetDefaultAtmRad():default の原子半径を設定する
SetDefaultColor():default の原子色を設定する
SetDefaultVdwRad():default の vdw 半径を設定する

```

## 6-4 fuView class (fuvview.py)・・・分子モデルの描画

### 1) attributes

```

getatmpos=True
raspos=[] # raster positon of aoms
rasposz=[] # z-position
#
updated=True
# the following data are used for setting center and draw size
atomdata=[]

```

```

bonddata=[]
chaintubedata=[]
drawtube=[]
# initial setting
eyepos = [0.0, 0.0, 300.0]
center = [0.0, 0.0, 0.0]
upward = [0.0, 1.0, 0.0]
ratio = fuView.DEFAULT_RATIO # angstrom per pixel
DisplayList = None
fog = True
fogscale=5.0
# default parameters
bgcolor = fuView.DEFAULT_BGCOLOR
rad_stick = fuView.DEFAULT_RAD_STICK
rad_ball = fuView.DEFAULT_RAD_BALL
rad_cpk_scale = fuView.DEFAULT_RAD_CPK_SCALE
rad_peptide = fuView.DEFAULT_RAD_PEPTIDE
stereo = fuView.STEREO_OFF
# flags for draw object
atom=False
bond=False
chaintube=False
selectcircle=False
selectrectangle=False
labelelm=False
labelatm=False
labelres=False
labelfrg=False
bdapoint=False
formalchg=False
distance=False
vdwbond=False
sphere=False

```

## 2) methods

```

CenterMolecular():
ClearScreen():

```

```
DrawAtoms (data) :
DrawBDAPoint (drawbdadata) :
DrawBonds (data) :
DrawChainTube (data) :
DrawDistance (data) :
DrawExtraBond (data) :
DrawLabel (drawdata) :
DrawSelectCircle () :
DrawSelectRectangle () :
DrawSphere (data) :
DrawText2 (text, font, pos, color) :
DrawText3 (text, font, pos, color) :
FitMolecular () :
GetAtomRasterPosition () :
    ret=raspos
GetCenter () :
    ret=(x, y, z)
GetFogScale () :
    ret=self.fog, self.fogscale
GetObjectXYZ (posx, posy) :
    ret=x, y, z
InitGL () : initialize OpenGL
MakeDisplayList () : make display list
MouseRotate (dif) :
MouseTranslate (dif) :
OnDraw () :
OnEraseBG () :
OnPaint () :
OnResize () :
SetBGColor () :
SetCamera () :
SetDistanceList (data) :
    ret=distance
SetDrawAtomData (on, data) :
SetDrawAtomList (on, data) :
SetDrawBDAPointData (on, data) :
```

```

SetDrawBDAPonitList (on, data) :
SetDrawBondData (on, data) :
SetDrawBondList (on, data) :
SetDrawChainTubeData (on, data) :
SetDrawChainTubeList (on, data) :
SetDrawDistanceData (on, data) :
SetDrawDistanceList (on, data) :
SetDrawFormalChargeData (on, data) :
SetDrawFormalChargeList (on, data) :
SetDrawLabelAtmData (on, data) :
SetDrawLabelAtmList (on, data) :
SetDrawLabelElmData (on, data) :
SetDrawLabelElmList (on, data) :
SetDrawLabelFrgData (on, data) :
SetDrawLabelFrgList (on, data) :
SetDrawLabelList (data) :
    ret=labeldata
SetDrawLabelResData (on, data) : set residue lable in
    DrawLabelData class form
    on(True/False) : draw on/off flag
    data (list:label,cc,color) : residue name data
SetDrawLabelResList (on, data) : set residue lable in list
    on(True/False) : draw on/off flag
    data (list) :
SetDrawNetChargeData (on, data) :
SetDrawNetChargeList (on, data) :
SetDrawSphereData (on, data) :
SetDrawSphereList (on, data) :
SetDrawVdwBondData (on, data) :
SetDrawVdwBondList (on, data) :
SetFogScale (on, fogscale) : flag and scale of fog
    on(True/False) : fog on/off
    fogscale(float) : fog scale, default=5.0
SetStereoCamera ( bCameraLeft, bViewportLeft) :
SetStereoView (stereo) : stereo view flag
    stereo (True/False)

```

```

SetViewAxis(xyz): set view axis
    xyz(string, 'X', 'Y' or 'Z'): view axis
Zoom(rot): magnify molecular model drawing
    rot(int): degree of magnification
Code2RGB(code)@static method: convert OpenGL RGB code to
    Windows code
    ret=RGB(list[r,g,b])
RGB2Code( RGB )@static method: convert Windows RGB code to OpenGL
    code
    ret=rgb(list[r,g,b])

```

6-5 fuCtrlFlag class (fuctrl.py)・・・プログラム制御 flags の保持。変更

1) attributes

```
ctrlflag={} # flag dictionary
```

2) methods

```
GetCtrlFlag(name):
```

```
    name(string): flag name
```

```
    ret=(True/False)
```

```
RemoveCtrlFlag(name): delete flag
```

```
    name(string): flag name
```

```
SetCtrlFlag(name,value): set flag value
```

```
    name(string): flag name
```

```
    value(True/False)
```

6-6 fuPlot class(fuplot.py)・・・fuplot の main program, file open

1) methods

```
About():
```

```
AddDerivedDataDic(drvnam,drvcmp):
```

```
CheckDeriveComp(drvcmp):
```

```
    ret=find
```

```
ConsoleMessage(mess):
```

```
ConvertCTChargeForPlot(fmoctchg): static method
```

```
    ret=ctcharge
```

```
ConvertGMSMullikenChargeForPlot(mulliken): static method
```

```
    ret=mulchg
```

```
ConvertMullikenChargeForPlot(fmomulchg): static method
```

```

    ret=mulchg
ConvertOneBodyForPlot(fmoone,layer): static method
    ret=onbody
ConvertPIEDAForPlot(fmopieda): static method
    ret=pieda,ctchg
CreateMatPlotLibFrame(): create MatPlotLib frame
CreatePropertyChoisePanel():create PropertyChoisePanel
CreatePyCrustFrame(): create PyCrust frame
CreateSelectDataPanel():create SelectDataPanel
CreateSplitWindow(): create split window for SelectDataPanel and
    PropertyChoisePanel
GetCurrentFMOData(): get current FMOProperty instance
    ret=curfmodat(FMOProperty instance): current FMO data
GetFMOProp(dataname): get FMOProper instance
    dataname(string): data name
    ret=fmodat(FMOProperty instance): FMO data of dataname
GetFMOPropName(fmodatadic,id):
    ret=dataname
GetIDAndName(dataname):
    ret=id,name
GetNameAndExt(filename): extract name and extension from file
    name
    filename(string): file name
    ret=err (True/False)
        ,name (string): name
        ,ext(string): extention
GetOpenFiles(curdir,files): get open multiple files
    curdir (string): current directory name
    files (list): file name list
IsDerivedData(dataname): is this derive data?
    dataname(string): plot data name
    ret=ret(True/False)
IsDuplicateName(dset,name): is name duplicate?
    dset(list): data name list
    name(string): name to check duplicate
    ret=dup(True/False)

```

```

IsFMOProperty(dataname): is data FMO property?
    dataname(string): data name
    ret=find(True/False)
IsItemInDataDic(item,datadic): is data in datadic?
    item(string): item name
    datadic(dictionary): data dictionary
    ret=ret(True/False)
ListFMODataName(): print FMO data name in fmodatadic dictionary
MakeCTChargePlotData():
    ret=ctcharge
MakeDataList():
    ret=datalist
MakeDataName(name):
    ret=dataname
MakeFMOPropertyDic(curdir,files):
    ret=fmodatadic
MakeMullikenPlotData():
    ret=mulcharge
MakePIEDAPlotData():
    ret=pieda
Message():
PlotMenu():
PrintFragmentName():
PrintMessage():
@staticmethod
ReadFMOCTCharge(filename): static method
    ret=err,version,ctchg
ReadFMOFragStatics(filename): static method
    ret=frgstatlist
ReadFMOInput(filename): static method
    ret=err,nfrag,icharg,frgnam,indatx,bdabaa
ReadFMOIterEnergy(filename): static method
    ret=err,version,jter,denergy,ddensity
ReadFMOKeywrd(filename,keywrd,col1,col2,datcollst): static
method
    ret=err,version,valuedic

```

```

ReadFMOMulliken(filename): static method
    ret=err,version,mulchg
ReadFMOOneBody(filename): static method
    ret=err,version,onebody
ReadFMOPIEDA(filename): static method
    ret=err,version,pieda
ReadFMOSTatics(filename): static method
    ret=err,version,prpdic
ReadFMOXYZ(filename): static method
    ret=err,geom
ReadFrgDistance(filename): static method
    ret=err,version,frgdist
ReadFrgIterEnergy(filename): static method
    ret=err,version,jter,efmo,dele,deld
ReadGSMulliken(filename): static method
    ret=err,version,mulchg
ResolveDerivedData(drvnam):
    ret=fmodat,cmpsign
RunMethod(method):
SavePropChoise(on):
SetDataListInSellB():
SetGraphData(pltprp):
SetPropChoise():
Version():
WriteRemark():

```

## 6 – 7 FMOProperty class (fuplot.py)...FMO output data

### 1) attributes and their default values

```

name=name # data name
outfile=outfil # FMO output file name
infile=inpfil # FMO input file name
pdbfile=pdbfil # pdb file name
# property flags
geom=3 # 0:no, 1:output file, 3:input file
pieda=False # =0:pie, =1:pieda
ctchg=False # =0:no data, 1:yes

```



```

mulchg=False # 0:not available, 1*yes
espot=False
density=False
orbital=False
# method in FMO
fmo=True
nbody=2 # # 2:fmo2, 3:fmo3
corr=0 # flag 0:hf, 1:corr
# property value
prpdic={} # dictionary of FMO property
etfmo2=0 # fmo2,fmo3
etfmo3=0
ecfmo2=0 # fmo2,3
ecfmo3=0 #
# molecule data
natmfrg=[] # natm in each fragment
natm=0 # total natm
nbas=0
nfrg=0
tchg=0 # total charge of while system
nbasfrg=[] # each fragment
frgnam=[] # fragment name
frgchg=[] # fragment charge
indat=[] # indat in FMO input data
bdabaa=[] # bda and baa
jobtitle='' # job comment in output
gamessver='' # GAMESS veision
fmover='' # FMO version in GAMESS
# FMO properties list
frgpieda=[] # pieda data
frgdist=[] # interfargment distance
ctcharge=[] # CT charge
mulliken=[] # mulliken population
onebody=[] # one-body energy

```

## 2) methods

```

SetAttributes(): read FMO attributes in FMO output and set them

```

to the attributes

```
GetFMOIterEnergy():  
    ret=iter,de,dd  
GetFrgIterEnergy():  
    ret=iter,efmo,de,dd
```

6-8 fuGraph class(fugraph) . . . controle graph drawing

1) methods

```
CreateChildGraphPanel(): create child graph panel  
CreateCTChargeCmdPanel(): create controle panel for CT charge plot  
CreateGraphPanel(): create panels(CreateCTChargeCmdPanel,  
    CreateMullikenCmdPanel,CreatePIEDACmdPanel)  
CreateMullikenCmdPanel(): create controle panel for Mulliken plot  
CreatePIEDACmdPanel(): create controle panel for PIEDA plot  
DrawChildMolView(): draw molecular model colored by property  
DrawGraph(): draw graph  
GetNumberOfPIEDAComponents(): get number of piedad components  
    ret=ncmp(int): number of piedad components  
GetRankColor(value): get color of function value  
    value(float): function value  
    ret=color(list): color  
MakeAllCTChargeData(): make plot data for all CT charge  
    ret=pltdat(list)  
MakeAllMullikenData():  
    ret=pltdat(list)  
MakeDataFor2D():  
    ret=pltdat2d(list)  
MakeFragmentCTChargeData(ifrg):  
    ret=pltdat(list)  
MakeFragmentMullikenData(ifrg):  
    ret=pltdat(list)  
MakeFragmentPIEDADData(ifrg): fragment の PIEDA データを作成する  
    ret=pltdat(list)  
MakePIEDABondingEnergyData():  
    ret=pltdat(list)  
MakePIEDARemarkList(pltdat):
```

```

    ret=lst(list)
MakeRankColorData():
MakeRemarkData():
    ret=remarkdata(list)
MenuGraph():
PrintData(idat):
PrintMaxData():
PrintMinData():
SetFMOProp(pltprp,datnam,molint,piedacmp,mullbody):
SetFMOPropData(natm,nfrg,frgnam,fmopr,frgdist):
SetMolViewFragmentData(pdbfile,indat,bdabaa):
SetMullikneBodyWedget(mullbody):
SetPIEDAComponentWedget(piedacmp): PIEDA components の wedget を設定
SetRankColor():Set Rank color
SetWedgetStates():Set wedget states
SortDataNmb(pltdat): Sort data in ascending data number
    ret=order
SortDist(): Sort data in acsending distance
    ret=order
SortLarge(pltdat): Sort data in descending order
    ret=order
SortSmall(pltdat): Sort data in acsending order
    ret=order

```

#### 6-9 BarGraph class(fugraph.py)・・・ Draw bar graph

##### 1) attributes and their defaults

```

    fontsize=[6,10] # font size in pixel
    fontcolor='black' # font color in wx.color
    # set graph size
    wplt=self.size[0]; hplt=self.size[1] # plot panel size
    # title and axis labels
    title='' # title text
    xlabel='' # title for x-axis
    ylabel='' # title for y-axis
    # plot data
    ndata=0 # number of data

```

```

data=[] # plot data
order=[] # order of data in plot
itemlist=self.SetDefaultItemList() # item list
extradata=False # only 1 extra data is allowed
extradatalabel='' # label of extra data
extposivalue=0.0
maxplotdata=0 # number of plot data; ndata or ndata+1(extra)
begindata=0 # the first sequence number of data to plot
focus=1 # seq. number of data to be drawn with thick frame
# remark
remarklist=[] # remark list
remarkboxsize=[10,8] # tile:[width,height]
wremark=0 # 10 width of remark
rank=10 # color rank
# y-axis range
wytitle=30 # width of y-title
wylabel=50 # width of y-label
xunit=12 # unit of x-axis in pixce,barwidth*2+2
barwidth=5 # bar width
hxtitle=20 # x axis title hight
hxlabel=30 # hight of x-label
# title
htitle=25 # title hight
#y range
yrangemin=-50.0 # +/- y max value
yunit=0 # =(self.yinipos-self.yendpos)/(2.0*self.yrange)
# scale out value
scaleoutposi=0.0; sscaleoutnega=0.0

```

## 2) methods

```

ClearGraph(): clear graph
DrawTitle(dc): draw graph title
    dc:device context
DrawAxisLabel(dc): draw x-label
    dc: device context
DrawRemark(dc): draw color remark

```

```

    dc: device context
DrawXAxis(dc):draw x-axis
    dc: device context
DrawYAxis(dc):draw y-axis
    dc: device context
FindXpos(value): graph frame の x position を返す
    value: x value to plot
    ret=x
FindYpos(value): graph frame の y position を返す
    value: y value to plot
    ret=y
Plot(on):graph を描画する
    on: True(draw) or False(clear)
PlotItemsInStackBar(dc,ii,x):
    dc:device context
    ii:int, ii 番目のデータを plot
    x:float, plot する x 軸の位置
Replot(xmove): mouse drag で図を scroll させる場合の再描画を行う
    xmove: int, mouse 移動量
SetAxisLabel(xlabel,ylabel): set a and y axis labels
    xlabel: string
    ylabel: string
SetBackgroundColor(color): set background color
    color: [r,g,b] or wx color, like 'white'
SetDefaultItemList(): set remark list for items to plot

ret=itemlist[['1',c1],['2',c2],['3',c3],['4',c4],['5',c5],['6',c6]]
SetData(data,order): set data and plot order
    data:[[0,value1],[1,value2],...]
    order:[0,i,2,j,...]
SetFont(font,fontsize): set font and its size
    font: wx.Font
    fontsize: list [width,height in pxcel]
SetItemNameAndColor(itemlist):
    itemlist:list [['text1',color1],['text2',color2],...]
SetPlotSize(wplt,hplt): set graph size

```

```

wplt(int): width in pixel, useally equal to panel width.
hplt(int): hight in pxcel, usually equal to panel hight
SetRemark(remarkboxsize,remarklist):
SetRankColor(negacolor,posicolor,negascaleoutcolor,
              posiscaleoutcolor):
SetTitle(title): set graph title
title(string):
SetYRange(ymin,ymax): set y-axis range
ymin(integer):
ymax(integer):

```

6-10 TileGraph class(fugraph.py) . . . draw tile graph

1) attributes and their defaults

```

focus=0 # focused fragmnet number
bgcolor=bgcolor
font = wx.Font(10, wx.FONTFAMILY_DEFAULT, wx.FONTSTYLE_NORMAL,
              wx.FONTWEIGHT_NORMAL, False, 'Courier 10 Pitch')
fontsize=[6,8]
fontcolor='black'
# graph size
wplt=self.size[0] # width of graph
hplt=self.size[1] # hight of graph
# title and axis labels
title='' # title text
xlabel='' # title for x-axis
ylabel='' # title for y-axis
# plot data
ndata=0 # number of data
data=[] # [[1,1,value,flag],...[1,n,value,flag]],
        # [[2,1,value,flag],...],.
order=[] # order of data in plot
extradata=False # only 1 extra data is allowed
extradatalabel='' # label of extra data
extposivalue=0.0 # scale out positive value
extnegavalue=0.0 # scale out negative value
maxplotdata=0 # number of plot data; ndata or ndata+1(extra)

```

```

xbegindata=0 # the first sequence number of data to plot
ybegindata=0 # the first sequence number of data to plot
# remark
remarklist=[] # list of remark
remarkboxsize=[8,8] # remark box size
remarktext='' # remark text
hremark=20 # hight of remark
rankposi=10 # rank fo rpositive value
ranknega=10 # rank for negative value
# rank color
posicolor='red' # will be converted to RGB255
negacolor='blue' # will be converted to RGB255
extnegacolor=[0,255,255] # color for scale out negative value,
# 'cyan'
extposicolor=[255,0,152] # color for scale out positive
# value,'magenta'
colorcode1=[255,255,0] # diagonal block color,'yellow'
colorcode2=[255,204,0] # covalent-bonded block color, 'gold'
tilesize=[10,10] # tile size
# x-axis range
wytittle=30 # width of y-title
wylabel=50 # width of y-label
wremark=55 # width of remark
xunit=12 # barwidth*2+2
barwidth=5 # width of bar
hxtittle=20 # hight of x-title
hxlabel=30 # hight of x-label
# y-axis range
htittle=25 # title hight
yrangemin=-50.0
yrangemax=50.0 # +/- y max value
yrange=self.yrangemax # y value range
scaleoutposi=0.0; scaleoutnega=0.0

```

2)methodos

```
ClearGraph():
```

```

DrawAxisLabel (dc) :
DrawRemark (dc, x, y, remarktext) :
DrawTitle (dc) :
FindXpos (value) :
    ret=x
FindYpos (value) :
    ret=y
GetColor (val, code) :
    ret=color
MakePlotData () :
    ret=plotdata
Plot (on) :
Replot (xmove, ymove) :
SetAxisLabel (xlabel, ylabel) :
SetBackgroundColor (color) :
SetBeginDataNumber (number) :
SetData (data, order) :
SetFocus (focus) :
SetFont (font, fontsize) :
SetItemNameAndColor (itemlist) :
SetPlotSize (wplt, hplt) :
SetRankColor (negacolor, posicolor, extnegacolor, extposicolor,
SetRemark (remarkboxsize, remarklist) :
SetRemarkTitle (text) :
SetTitle (title) :
SetYRange (yrangemin, yrangemax) :

```

6-11 MatPlotLib\_Frm class (fugraph.py) ...Frame for MatPlotLib graph

1) methods

```

Clear(): clear graph
NewPlot(): create subplot (figure.add_subplot(111))
PlotTitle(text): title をプロットする
    text(string): title text
PlotXLabel(text):plot x-axis label
    text(string): label text
PlotXY(x,y): plot (x,y) deta

```



```

    x(list,[1,2,...]): list of x values
    y(list,[1,2,...]): list of y values
    PlotYLabel(text): plot y-axis label
    text(string): label text

```

## 6-12 fulib modules (fulib.py) . . . Global methods

### 1) methods

```

AngleT(ra,rb): calculate angle between two vectors
    ra(list,[x,y,z]): point vector
    rb(list,[x,y,z]): point vector
    ret=t(float): angle in radian
AtmNamElm(atmnam): get element from PDB atom name
    atmnam(string*4): PDB atom name
    ret=elm(string*2): element
ChangeToPreviousDir(dirname,inifile): get directory name in
    inifile and change current directory to it
    dirname(string): directory of the executable program
    inifile(string): file name in which directory name is kept
    ret=dir(string): current directory name
ChooseColorPanel(parent): open color picker panel
    parent(frame instance): parent frame
CofMassPmi(atmas,coord):
    ret=com.vec
CompressIntData(expnint):
    ret=cmpint
CopyBitmapToClipboard(bmp):
CovalentBondedAtomList(elm,coord):
    ret=bndlst
DihedralAngle(iatm,jatm,katm,latm):
DihedralAngleMainChain():
DihedralAngleOfCA(ires,jres,kres,lres):
Distance(p1,p2): calculate distance between two points
    p1(list[x,y,z]): point coordinate
    p2(list[x,y,z]): point coordinate
    ret=r(float): distance between p1 and p2
ExpandIntData(cmpint): expand packed integers

```

```

    compint(list):
        ret=expnint(list)
ExtractWaters(molnam,pdbmol):
    ret=watdat,moldat
GetExeDir(program): get directory of executable program
    program(string): program name
    ret=dir(string): directory
HSVtoRGB(HSVcol): convert HSV color code to RGB code
    HSVcol(list): HSV color
    ret=RGBcol(list): RGB color
ListPDBResSeq(pdbmol,skipter,skipwat):
    ret=resnamlst
OrientOrg(pdborg,pdbdrv):
    ret=pdbdrv
PrintNumericList(filnam,intlst,header,sep,width,colnw):
PrintStringList(filnam,strlst,header,sep,width,colnw):
RotMatAxi(v,t): return matrix u for rotation by t (radian) around
                vector v.
    v(list): axis vector
    t(float): angle in radian
    ret=u(list): (3x3) transformation matrix
RotMatEul(a,b,c): return rotation matrix u for (a,b,c) Euler angle
    a(float): angle alpha
    b(float): angle beta
    c(float): angle gamma
    ret=u(list): (3x3) transformation matrix
RotMatPnts(refpnt,newpnt): get rotation matrix transforming
                refrence points to new points
    refpnt(list): reference points coordinates
    newpnt(list): new coordinates
    ret=u(list): (3x3) transformation matrix
RotMatX(a):
    ret=ux(list): (3x3) transformation matrix
RotMatY(b):
    ret=uy(list): (3x3) transformation matrix
RotMatZ(c):

```

```

    ret=uz(list): (3x3) transformation matrix
RotMol(u,cnt,coord):座標を変換する
    1) u(list[[],[],[ ]]): transformation matrix
        cnt(list[x,y,z]): center of rotation
        coord(list[x,y,z]):, coordinates
    2) xyz(list[x,y,z]): transformed coordinates
SplitAtTER(mnam,pdbmol):
    ret=molnamdic,moldatdic,delcondic
SplitRes(molnam,pdbmol):
    ret=molnamdic,moldatdic
StringToInteger(strdata):
    ret=intdata
TrnMat(ra,ri,rf): get transformation matrix to rotate ri to rf
                    around axis ra
    ra(list[x,y,z]): axial vector for rotation
    ri(list[x,y,z]): initial position vector
    rf(list[x,y,z]): final position vector
    ret=u(list): transformation matrix
WriteDirectoryOnFile(dirname,filename): write directry name on a
                                        file
    dirname(string):directory name
    filename(string): file name
    ret=none

```

### 6-13 Class in fupanel.py module

1) classes of pop-up panels in fu.

```

class ControlPanel_Frm(wx.Frame): control panel
class DeriveDataInput_Frm(wx.Frame): derived data input panel
class GamessInput_Frm(wx.Frame): GAMESS input assistant panel
class GroupJoin_Frm(wx.Frame): Group joint panel
class GroupRename_Frm(wx.Frame): Group rename panel
class InputAtomCharge_Frm(wx.Frame): input atom charge panel
class LayerAssignPanel_Frm(wx.Frame): layer assignment panel
class NameSelector_Frm(wx.Frame): Name/number select panel
class OpenMultipleFile_Frm(wx.Frame): Open multople files panel
class RadiusSelector_Frm(wx.Frame): Radius selection panel

```

```
class TreeSelector_Frm(wx.Frame): Tree selector panel
```

## 7. Miscellaneous

### 1) Sequence number of Atoms and 'TER' atom

It is noted that, in the program atoms are numbered from 0 and in I/O interface from 1. The treatment of 'TER's (PDB ATOM record) is tricky. The fu keeps all 'TER' without subjected in any operations except DeleteTerAtoms method of fuModel class. In fragmentation of a molecule, complicated codes are needed as in SetFragmentName method in fuMole class, since 'TER' has sequence number.

### 2) Types of AdHydrogen

Table AddHydrogenType (addhtype)

addhtype code	connection of reference atoms	comment
1A1	-atom2 H-atom1-atom3  -atom4	H atom and atom1-4 form pseudotetrahedron. ex. H-CH3 of methane
1A2	-atom2 H-atom1  -atom3	atom1 is sp2 and H and atom1-3 are in a plane. ex. H-C of benzene
1A3	H-atom1-atom2-atom3	H-atom1-atom2-atom3 can be cis or trans. ex. H-O-C-C of ethanol
2A1	H-   -atom2 atom1 H-   -atom3	two H atoms are added at atom1 H2 and atom2,3 are twisted by 90 degrees. ex. H2 of CH4
2A2	H-  atom1-atom2-atom3 H-	all atoms are in a plane ex. H2N-C-N(H2)- of arginine

```
3A1      H-|                               three H atoms are added at atom1
          H-atom1-atom2-atom3             ex. H3N-C-C- of lysine
          H-|
```

---

### 3) tutorials

The following tutorials are attached to help programming with fu(fu/src/tutorial directory).

```
#tutorial_01: create window with StatusBar and Menu.
```

```
Practice: use of wxFrame, wxCreateStatusBar and fuMenu class
          in fuctrl.py module.
```

```
#tutorial_02: file I/O and PyCrust shell
```

```
Practice: 1) use of ReadPDBData method, a static method of fuModel
          class in fumodel.py module.
```

```
2) use of PyCrust
```

```
Usage:  1) Execute 'File'-'Open' command and read PDB file.
        2) Then type 'print demo.pdbmol[enter]' in PyCrust
          console.
```

```
3) Type 'demo.Methdo()[enter]' for using attribute of
   'demo' (an instance of Tutorial_02 class).
```

```
Note: You see that you can access attributes and methods in
      running GUI program through python program in PyCrust
      console. That means you can modify data, you can add
      functions interactively, without modifying the source
      codes of the program.
```

```
#tutorial_03: use of fuView class in fuvview.py module
```

```
Practice: Draw line model (or CPK model) of CH4 molecule.
```

```
Usage: The model is rotated and magnified by mouse move with
       left button down and mouse wheel rotation, respectively.
```

```
# tutorial_04: use of fuMole class in fumole.py with fuView class
             in fuvview.py module
```

```
Practice: Draw molecule created from PDB data.
```

Usage: The model is rotated and magnified by mouse move with left button down and mouse wheel rotation, respectively.

#tutorial\_05: BarGraph class in fugraph.py module

Practice: Draw bargraph.

Usage: When graph is sticked out the panel, mouse move with left button down scrolls the graph.

# tutorial\_06: TileGraph class in fugraph.py module

Practice: Draw tilegraph.

Usage: When graph is sticked out the panel, mouse move with left button down scrolls the graph.

#### 4) Icon for Windows7

Windows7 Icon(24bit color, a set of 48x48,32x32,24x24,16x16 pixcel size) was created by

IconWiz(<http://www.vector.co.jp/soft/dl/winnt/amuse/se430751.html>).

The original image (320x320 pixcel, 24bit color bitmap) was prepared using the paint of Windows7.