

NAME

MolecularDescriptors - MolecularDescriptors class

SYNOPSIS

```
use MolecularDescriptors;

use MolecularDescriptors qw(:all);
```

DESCRIPTION

MolecularDescriptors base class used to derive all other molecular descriptors classes provides the following methods:

`new`, `AddDescriptorNames`, `AddDescriptorValues`, `GetDescriptorNames`, `GetDescriptorNamesAndValues`, `GetDescriptorValueByName`, `GetDescriptorValues`, `IsDescriptorsGenerationSuccessful`, `SetDescriptorNames`, `SetDescriptorValues`, `SetMolecule`, `SetType`

MolecularDescriptors class is derived from `ObjectProperty` base class which provides methods not explicitly defined in `Fingerprints` or `ObjectProperty` classes using Perl's AUTOLOAD functionality. These methods are generated on-the-fly for a specified object property:

```
Set<PropertyName>( <PropertyValue> );
$PropertyValue = Get<PropertyName>();
Delete<PropertyName>();
```

METHODS

`new`

```
$NewMolecularDescriptors = new MolecularDescriptors(%NamesAndValues);
```

Using specified *MolecularDescriptors* property names and values hash, `new` method creates a new object and returns a reference to newly created `MolecularDescriptors` object. By default, following properties are initialized:

```
Molecule = '';
Type = '';
```

`AddDescriptorNames`

```
$MolecularDescriptors->AddDescriptorNames(@Name);
```

Adds specified descriptor *Names* to the list of available descriptor names and returns *MolecularDescriptors*.

`AddDescriptorValues`

```
$MolecularDescriptors->AddDescriptorValues(@Values);
```

Adds specified descriptor *Values* to the list of calculated descriptor values and returns *MolecularDescriptors*.

`GetDescriptorNames`

```
@Names = $MolecularDescriptors->GetDescriptorNames();
```

Returns an array containing all available descriptor names.

`GetDescriptorNamesAndValues`

```
%NamesAndValuesReturn = $MolecularDescriptors->
    GetDescriptorNamesAndValues();
```

Returns a hash containing all available descriptor names and calculated values.

`GetDescriptorValueByName`

```
$Value = $MolecularDescriptors->
    GetDescriptorValueByName($Name);
```

Returns calculated value for a specified descriptor name. A string `None` is returned for unknown descriptor names or for those descriptors whose values haven't been calculated.

`GetDescriptorValues`

```
@Values = $MolecularDescriptors->GetDescriptorValues();
```

Returns an array containing calculated descriptor values for all available descriptors. Unless `CalculateDescriptorsValues` method has been successfully invoked on a *MolecularDescriptors* object, value of each descriptor corresponds to string `None`.

`IsDescriptorsGenerationSuccessful`

```
$Status = $MolecularDescriptors->
    IsDescriptorsGenerationSuccessful();
```

Returns 1 or 0 based on whether molecular descriptors generation was successful. For a successful molecular descriptors calculation, all available descriptors must have a values other than a string *None* which are set by CalculateDescriptorsValues method after successful completion of descriptors calculation.

SetDescriptorNames

```
$MolecularDescriptors->SetDescriptorNames(@Names);
```

Sets names of available descriptors to specified names and returns *MolecularDescriptors*.

SetDescriptorValues

```
$MolecularDescriptors->SetDescriptorValues(@Values);
```

Sets values of available descriptors to specified values and returns *MolecularDescriptors*.

SetMolecule

```
$MolecularDescriptors->SetMolecule($Molecule);
```

Sets molecule to use during calculation of molecular descriptors and returns *MolecularDescriptors*.

SetType

```
$MolecularDescriptors->SetType($Type);
```

Sets *Type* for *MolecularDescriptors* object and returns *MolecularDescriptors*.

AUTHOR

Manish Sud <msud@san.rr.com>

SEE ALSO

MolecularDescriptorsGenerator.pm

COPYRIGHT

Copyright (C) 2004-2012 Manish Sud. All rights reserved.

This file is part of MayaChemTools.

MayaChemTools is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 3 of the License, or (at your option) any later version.