NAME

TPSADescriptors

SYNOPSIS

use MolecularDescriptors::TPSADescriptors;

use MolecularDescriptors::TPSADescriptors qw(:all);

DESCRIPTION

TPSADescriptors class provides the following methods:

new, GenerateDescriptors, GetDescriptorNames, StringifyTPSADescriptors

TPSADescriptors is derived from MolecularDescriptors class which in turn is derived from ObjectProperty base class that provides methods not explicitly defined in TPSADescriptors, MolecularDescriptors 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>();
```

After Topological Polar Surface Area (TPSA) atom types [Ref 90-91] has been assigned to appropriate atoms in a molecule using AtomTypes::TPSAAtomTypes.pm module, TPSA value is calculated by adding up contributions of each appropriate atom type.

By default, MayaChemTools only uses nitrogen and oxygen atoms during calculation of TPSA and ignores phosphorous and sulfur atoms. [Ref 90 - 91]

METHODS

new

```
$NewTPSADescriptors = new MolecularDescriptors::TPSADescriptors(
                                                    %NamesAndValues);
```

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

```
Molecule = ''
Type = 'TPSA'
IgnorePhosphorus = 1
IgnoreSulfur = 1
@DescriptorNames = ('TPSA')
@DescriptorValues = ('None')
```

Examples:

```
$TPSADescriptors = new MolecularDescriptors::TPSADescriptors(
                           'Molecule' => $Molecule);

$TPSADescriptors = new MolecularDescriptors::TPSADescriptors();

$TPSADescriptors = new MolecularDescriptors::TPSADescriptors(
                           'IgnorePhosphorus' => 0,
                           'IgnoreSulfur' => 0);

$TPSADescriptors->SetMolecule($Molecule);
$TPSADescriptors->GenerateDescriptors();
print "TPSADescriptors: $TPSADescriptors\n";
```

GenerateDescriptors

```
$TPSADescriptors->GenerateDescriptors();
```

Calculate TPSA value for a molecule and returns *TPSADescriptors*.

GetDescriptorNames

```
@DescriptorNames = $TPSADescriptors->GetDescriptorNames();
@DescriptorNames = MolecularDescriptors::TPSADescriptors::
                   GetDescriptorNames();
```

Returns all available descriptor names as an array.

StringifyTPSADescriptors

```
$String = $TPSADescriptors->StringifyTPSADescriptors();
```

Returns a string containing information about *TPSADescriptors* object.

AUTHOR

Manish Sud <msud@san.rr.com>

SEE ALSO

MolecularDescriptors.pm, MolecularDescriptorsGenerator.pm

COPYRIGHT