

## NAME

PeriodicTable

## SYNOPSIS

use PeriodicTable;

use PeriodicTable qw(:all);

## DESCRIPTION

PeriodicTable module provides the following functions:

GetElementMostAbundantNaturalIsotopeData, GetElementMostAbundantNaturalIsotopeMass, GetElementMostAbundantNaturalIsotopeMassNumber, GetElementNaturalIsotopeAbundance, GetElementNaturalIsotopeCount, GetElementNaturalIsotopeMass, GetElementNaturalIsotopesData, GetElementPropertiesData, GetElementPropertiesNames, GetElementPropertiesNamesAndUnits, GetElementPropertyUnits, GetElements, GetElementsByAmericanStyleGroupLabel, GetElementsByEuropeanStyleGroupLabel, GetElementsByGroupName, GetElementsByGroupNumber, GetElementsByPeriodNumber, GetIUPACGroupNumberFromAmericanStyleGroupLabel, GetIUPACGroupNumberFromEuropeanStyleGroupLabel, IsElement, IsElementNaturalIsotopeMassNumber, IsElementProperty

## METHODS

## GetElements

```
@ElementSymbols = GetElements();
$ElementSymbolsRef = GetElements();
```

Returns an array or a reference to an array of known element symbols

## GetElementsByGroupName

```
@ElementSymbols = GetElementsByGroupName($GroupName);
$ElementSymbolsRef = GetElementsByGroupName($GroupName);
```

Returns an array or a reference to an array of element symbols for a specified *GroupName*. Supported *GroupName* values are: *Alkali metals*, *Alkaline earth metals*, *Coinage metals*, *Pnictogens*, *Chalcogens*, *Halogens*, *Noble gases*; Additionally, usage of *Lanthanides* (Lanthanoids) and *Actinides* (Actinoids) is also supported.

## GetElementsByGroupNumber

```
@ElementSymbols = GetElementsByGroupNumber($GroupNumber);
$ElementSymbolsRef = GetElementsByGroupNumber($GroupNumber);
```

Returns an array or a reference to an array of element symbols for a specified *GroupNumber*

## GetElementsByAmericanStyleGroupLabel

```
@ElementSymbols = GetElementsByAmericanStyleGroupLabel($GroupLabel);
$ElementSymbolsRef = GetElementsByAmericanStyleGroupLabel($GroupLabel);
```

Returns an array or a reference to an array of element symbols for a specified American style *GroupLabel*. Valid values for American style group labels: *IA to VIIIA*, *IB to VIIIB*, *VIII*.

## GetElementsByEuropeanStyleGroupLabel

```
@ElementSymbols = GetElementsByEuropeanStyleGroupLabel($GroupLabel);
$ElementSymbolsRef = GetElementsByEuropeanStyleGroupLabel($GroupLabel);
```

Returns an array or a reference to an array of element symbols for a specified European style *GroupLabel*. Valid values for European style group labels: *IA to VIIIA*, *IB to VIIIB*, *VIII*.

## GetElementsByPeriodNumber

```
@ElementSymbols = GetElementsByPeriodNumber($PeriodNumber);
$ElementSymbolsRef = GetElementsByPeriodNumber($PeriodNumber);
```

Returns an array or a reference to an array of element symbols for a specified *PeriodNumber*.

## GetElementMostAbundantNaturalIsotopeData

```
@IsotopeData = GetElementMostAbundantNaturalIsotopeData(
    $ElementID);
$IsotopeDataRef = GetElementMostAbundantNaturalIsotopeData(
```

```
$ElementID);
```

Returns an array or reference to an array containing data for most abundant isotope of an element specified by element symbol or atomic number. Isotope data arrays contain these values: *AtomicNumber*, *IsotopeSymbol*, *MassNumber*, *RelativeAtomicMass*, and *NaturalAbundance*.

GetElementMostAbundantNaturalIsotopeMassNumber

```
$MassNumber = GetElementMostAbundantNaturalIsotopeMassNumber($ElementID);
```

Returns mass number of most abundant natural isotope of an element specified by element symbol or atomic number

GetElementNaturalIsotopeCount

```
$IsotopeCount = GetElementNaturalIsotopeCount($ElementID);
```

Returns natural isotope count for an element specified by element symbol or atomic number

GetElementNaturalIsotopesData

```
$DataHashRef = GetElementNaturalIsotopesData($ElementID,
[ $MassNumber ] );
```

Returns a reference to a hash containing all available isotope data for an element specified using element symbol or atomic number; an optional mass number indicates retrieve data for a specific isotope

GetElementNaturalIsotopeAbundance

```
$Abundance = GetElementNaturalIsotopeAbundance($ElementID,
$MassNumber);
```

Returns percent abundance of natural isotope for an element with specific mass number.

GetElementMostAbundantNaturalIsotopeMass

```
$RelativeAtomicMass = GetElementMostAbundantNaturalIsotopeMass(
$ElementID);
```

Returns relative atomic mass of most abundant isotope for an element specified using element symbol or atomic number.

GetElementNaturalIsotopeMass

```
$RelativeAtomicMass = GetElementNaturalIsotopeMass($ElementID,
$MassNumber);
```

Returns relative atomic mass of an element with specific mass number.

GetElementPropertiesData

```
$PropertyDataHashRef = GetElementPropertiesData($ElementID);
```

Returns a reference to a hash containing all available properties data for an element specified using element symbol or atomic number.

GetElementPropertyName

```
$PropertyValue = GetElement<PropertyName>($ElementID);
```

Returns value of an element for a element specified using element symbol or atomic number.

These functions are not defined in this modules; these are implemented on-the-fly using Perl's AUTOLOAD functionality.

Here is the list of known element *property names*: *AllenElectronegativity*, *AllredRochowElectronegativity*, *AtomicNumber*, *AtomicRadiusCalculated*, *AtomicRadiusEmpirical*, *AtomicWeight*, *Block*, *BoilingPoint*, *BondLength*, *BrinellHardness*, *BulkModulus*, *Classification*, *CoefficientOfLinearExpansion*, *Color*, *CommonValences*, *LowestCommonValence*, *HighestCommonValence*, *CommonOxidationNumbers*, *LowestCommonOxidationNumber*, *HighestCommonOxidationNumber*, *CovalentRadiusEmpirical*, *CriticalTemperature*, *DensityOfSolid*, *DiscoveredAt*, *DiscoveredBy*, *DiscoveredWhen*, *ElectricalResistivity*, *ElectronAffinity*, *ElementName*, *ElementSymbol*, *EnthalpyOfAtomization*, *EnthalpyOfFusion*, *EnthalpyOfVaporization*, *FirstIonizationEnergy*, *GroundStateConfiguration*, *GroundStateLevel*, *GroupName*, *GroupNumber*, *NaturalIsotopeData*, *MeltingPoint*, *MineralHardness*, *MolarVolume*, *MullikenJaffeElectronegativity*, *OriginOfName*, *PaulingElectronegativity*, *PeriodNumber*, *PoissonsRatio*, *Reflectivity*, *RefractiveIndex*, *RigidityModulus*, *SandersonElectronegativity*,

---

StandardState, SuperconductionTemperature, ThermalConductivity, VanderWaalsRadius, VelocityOfSound, VickersHardness, YoungsModulus.

#### GetElementPropertiesNames

```
@PropertyNames = GetElementPropertiesNames([$Mode]);  
$PropertyNamesRef = GetElementPropertiesNames([$Mode]);
```

Returns names of all available element properties. Optional mode parameter controls grouping of property names; Possible values: *ByGroup* or *Alphabetical*. Default: *ByGroup*.

#### GetElementPropertiesNamesAndUnits

```
$NameUnitsHashRef = GetElementPropertiesNamesAndUnits();
```

Returns a reference to a hash of property names and units of all available element properties. Names with no units contains empty strings.

#### GetElementPropertyUnits

```
$Units = GetElementPropertyUnits($PropertyName);
```

Returns units for a specific element property name. An empty string is returned for a property with no units.

#### GetIUPACGroupNumberFromAmericanStyleGroupLabel

```
$GroupNumber = GetIUPACGroupNumberFromAmericanStyleGroupLabel(  
$GroupLabel);
```

Returns IUPAC group numbers of a specific American style group label. A comma delimited string is returned for group VIII or VIII B.

#### GetIUPACGroupNumberFromEuropeanStyleGroupLabel

```
$GroupNumber = GetIUPACGroupNumberFromEuropeanStyleGroupLabel(  
$GroupLabel);
```

Returns IUPAC group numbers of a specific European style group label. A comma delimited string is returned for group VIII or VIII A.

#### IsElement

```
$Status = IsElement($ElementID);
```

Returns 1 or 0 based on whether it's a known element symbol or atomic number.

#### IsElementNaturalIsotopeMassNumber

```
$Status = IsElementNaturalIsotopeMassNumber($ElementID, $MassNumber);
```

Returns 1 or 0 based on whether it's a valid mass number for an element symbol or atomic number.

#### IsElementProperty

```
$Status = IsElementProperty($PropertyName);
```

Returns 1 or 0 based on whether it's a valid property name.

#### AUTHOR

Manish Sud <msud@san.rr.com>

#### SEE ALSO

AminoAcids.pm, NucleicAcids.pm

#### COPYRIGHT

Copyright (C) 2025 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.