

---

# **pyepsg Documentation**

*Release 0.4.0*

**Richard Hattersley**

**Nov 26, 2018**



---

## Contents

---

**Python Module Index**

**5**



Provides simple access to <http://epsg.io/>.

The entry point for this package is the `get ()` function.

**class** `pyepsg.Axis` (*element*)

Bases: `pyepsg.EPSG`

A single coordinate axis.

**direction**

A description of the orientation of this axis.

**uom**

The name of the unit of measure used on this axis.

**class** `pyepsg.CRS` (*element*)

Bases: `pyepsg.EPSG`

Abstract parent class for `GeodeticCRS`, `ProjectedCRS` and `CompoundCRS`.

**as\_esri\_wkt ()**

Return the ESRI WKT string which corresponds to the CRS.

For example:

```
>>> print(get(27700).as_esri_wkt())
PROJCS["OSGB_1936_British_National_Grid",GEOGCS["GCS_OSGB_19...
```

**as\_html ()**

Return the OGC WKT which corresponds to the CRS as HTML.

For example:

```
>>> print(get(27700).as_html())
<div class="syntax"><pre><span class="gh">PROJCS</span><span...
```

**as\_proj4 ()**

Return the PROJ.4 string which corresponds to the CRS.

For example:

```
>>> print(get(21781).as_proj4())
+proj=somerc +lat_0=46.95240555555556 +lon_0=7.439583333333333 +k_0=1 +x_
↪0=600000 +y_0=200000 +ellps=bessel +towgs84=674.4,15.1,405.3,0,0,0,0_
↪+units=m +no_defs
```

**as\_wkt ()**

Return the OGC WKT string which corresponds to the CRS.

For example:

```
>>> print(get(27700).as_wkt())
PROJCS["OSGB_1936 / British National Grid",GEOGCS["OSGB_1936...
```

**domain\_of\_validity ()**

Return the domain of validity for this CRS as: (west, east, south, north).

For example:

```
>>> print(get(21781).domain_of_validity())
[5.96, 10.49, 45.82, 47.81]
```

**id**

The EPSG code for this CRS.

**name**

The human-readable name.

**scope**

A human-readable description of the intended usage for this CRS.

**class** `pyepsg.CartesianCS` (*element*)

Bases: `pyepsg.EPSG`

A 1-, 2-, or 3-dimensional cartesian coordinate system.

**axes**

An ordered list of `Axis` objects describing X and Y.

**name**

The human-readable name.

**remarks**

Human-readable comments.

**class** `pyepsg.CompoundCRS` (*element*)

Bases: `pyepsg.CRS`

Represents a single compound CRS.

**class** `pyepsg.EPSG` (*element*)

Bases: `object`

Parent class of all objects returned by `pyepsg`.

**identifier**

The official URN for this object.

**class** `pyepsg.GeodeticCRS` (*element*)

Bases: `pyepsg.CRS`

Represents a single geodetic CRS.

**class** `pyepsg.ProjectedCRS` (*element*)

Bases: `pyepsg.CRS`

Represents a single projected CRS.

**base\_geodetic\_crs**

The `GeodeticCRS` on which this projection is based.

**cartesian\_cs**

The `CartesianCS` which describes the coordinate axes.

**class** `pyepsg.UOM` (*element*)

Bases: `pyepsg.EPSG`

A unit of measure.

**name**

The human-readable name.

`pyepsg.get` (*code*)

Return an object that corresponds to the given EPSG code.

**Currently supported object types are:**

- `GeodeticCRS`

- *ProjectedCRS*
- *CartesianCS*
- *UOM*

For example:

```
>>> print(get(27700))
<ProjectedCRS: 27700, OSGB 1936 / British National Grid>
>>> print(get('4400-cs'))
<CartesianCS: Cartesian 2D CS. Axes: easting, northi..>
>>> print(get(5973))
<CompoundCRS: 5973, ETRS89 / UTM zone 33 + NN2000 height>
```





**p**

pyepsg, ??



## A

as\_esri\_wkt() (pyepsg.CRS method), 1  
as\_html() (pyepsg.CRS method), 1  
as\_proj4() (pyepsg.CRS method), 1  
as\_wkt() (pyepsg.CRS method), 1  
axes (pyepsg.CartesianCS attribute), 2  
Axis (class in pyepsg), 1

## B

base\_geodetic\_crs (pyepsg.ProjectedImage attribute), 2

## C

cartesian\_cs (pyepsg.ProjectedImage attribute), 2  
CartesianCS (class in pyepsg), 2  
CompoundCRS (class in pyepsg), 2  
CRS (class in pyepsg), 1

## D

direction (pyepsg.Axis attribute), 1  
domain\_of\_validity() (pyepsg.CRS method), 1

## E

EPSG (class in pyepsg), 2

## G

GeodeticCRS (class in pyepsg), 2  
get() (in module pyepsg), 2

## I

id (pyepsg.CRS attribute), 1  
identifier (pyepsg.EPSG attribute), 2

## N

name (pyepsg.CartesianCS attribute), 2  
name (pyepsg.CRS attribute), 2  
name (pyepsg.UOM attribute), 2

## P

ProjectedCRS (class in pyepsg), 2

pyepsg (module), 1

## R

remarks (pyepsg.CartesianCS attribute), 2

## S

scope (pyepsg.CRS attribute), 2

## U

UOM (class in pyepsg), 2  
uom (pyepsg.Axis attribute), 1