

---

# **sopn-publish-date**

*Release 1.4.0*

**Jun 15, 2020**



---

# Contents

---

<b>1</b>	<b>Overview</b>	<b>1</b>
1.1	Usage . . . . .	1
1.2	Installation . . . . .	1
1.3	Third-Party Sources . . . . .	2
<b>2</b>	<b>Legislation</b>	<b>3</b>
2.1	United Kingdom . . . . .	3
2.2	England . . . . .	3
2.3	Wales . . . . .	3
2.4	Scotland . . . . .	3
2.5	Northern Ireland . . . . .	3
<b>3</b>	<b>Tests</b>	<b>5</b>
<b>4</b>	<b>API</b>	<b>7</b>
4.1	sopn_publish_date . . . . .	7
4.2	sopn_publish_date.calendars . . . . .	9
4.3	sopn_publish_date.election_ids . . . . .	10
4.4	sopn_publish_date.date . . . . .	11
	<b>Python Module Index</b>	<b>13</b>
	<b>Index</b>	<b>15</b>



*Given the polling day of an election in the UK, when should the Statement of Persons Nominated (SoPN) be published?*

When an election is called in the United Kingdom, the voting public must have access to the list of candidates who have been nominated to that post.

These documents are called Statements of Persons Nominated, and must be published a set number of working days ahead of the actual election date. The number varies based on:

- *Type of Post* - Parliamentary, Local, devolved Government, etc.
- *Country* - The United Kingdom has up to four different rules for the same type of election, one for each country.
- *Calendar* - each country has their own unique set of Bank Holidays.

This library encapsulates timetable legislation for elections run in the United Kingdom and its devolved administrations.

## 1.1 Usage

```
from sopn_publish_date import StatementPublishDate
from datetime import date

publish_date = StatementPublishDate()

print(publish_date.national_assembly_for_wales(date(2016, 5, 5)))

# datetime.date(2016, 4, 7)
```

## 1.2 Installation

```
pip install sopn_publish_date
```

## 1.3 Third-Party Sources

The bank holidays JSON used in this project is provided by [gov.uk](https://www.gov.uk) under the [Open Government Licence](https://www.ogilivys.com/insights/open-government-licence)

Below are the types of election supported by this library and their corresponding pieces of legislation.

### **2.1 United Kingdom**

### **2.2 England**

### **2.3 Wales**

### **2.4 Scotland**

### **2.5 Northern Ireland**





## CHAPTER 3

---

### Tests

---

Writing tests for code that represents legislation is a bit fiddly.

Ideally we'd have perfect test cases but as noted in *Overview* there are concrete examples of a candidate rejecting their nomination, pushing back the publishing of the SoPN papers.

This project has two sets of tests:

- Unit tests, with single-specified examples
- Approval tests, with test data sourced from parsing historic SoPNs (provided by [Democracy Club](#), who maintain a database of candidates and elections)



## 4.1 sopn\_publish\_date

`class sopn_publish_date.StatementPublishDate`

**europaean\_parliament** (*poll\_date: datetime.date, region: sopn\_publish\_date.calendars.Region*) → datetime.date

Calculate the publish date for an election to the European Parliament

This is set out in [The European Parliamentary Elections \(Amendment\) Regulations 2009](#)

As Gibraltar is included within *South West England* but has a different bank holiday calendar, it's plausible that the SoPN date for *South West England* will differ from the rest of England.

### Parameters

- **poll\_date** – a datetime representing the date of the poll
- **region** – the region of the UK and Gibraltar where the poll is being run

**Returns** a datetime representing the expected publish date

**for\_id** (*election\_id: str, country: sopn\_publish\_date.calendars.Country = None*) → datetime.date

Calculate the publish date for an election given in [uk-election-ids](#) format and an optional country if necessary (for example, local or parliamentary elections).

This function returns *None* for elections to the European Parliament, and will raise an exception if the election id is ambiguous (could correspond to elections in multiple countries with different electoral legislation).

### Parameters

- **election\_id** – a string representing an election id in [uk-election-ids](#) format
- **country** – an optional Country representing the country where the election will be held

**Returns** a datetime representing the expected publish date

**greater\_london\_assembly** (*poll\_date: datetime.date*) → datetime.date

Calculate the publish date for an election to the Greater London Assembly

This is set out in [The Greater London Authority Elections \(Amendment\) Rules 2016](#)

**Parameters** **poll\_date** – a datetime representing the date of the poll

**Returns** a datetime representing the expected publish date

**local** (*poll\_date: datetime.date, country: sopn\_publish\_date.calendars.Country*)

Calculate the publish date for a local election.

This is set out in:

- [The Local Elections \(Principal Areas\) \(England and Wales\) \(Amendment\) Rules 2014](#)
- [The Local Elections \(Northern Ireland\) Order 2010](#)
- [The Scottish Local Government Elections Order 2011](#)

**Parameters**

- **poll\_date** – a datetime representing the date of the poll
- **country** – the country in which the election is being run

**Returns** a datetime representing the expected publish date

**mayor** (*poll\_date: datetime.date*) → datetime.date

Calculate the publish date for an election to the position of Mayor in England and Wales

This is set out in [The Local Authorities \(Mayoral Elections\) \(England and Wales\) \(Amendment\) Regulations 2014](#)

**Parameters** **poll\_date** – a datetime representing the date of the poll

**Returns** a datetime representing the expected publish date

**national\_assembly\_for\_wales** (*poll\_date: datetime.date*) → datetime.date

Calculate the publish date for an election to the National Assembly for Wales

This is set out in [The National Assembly for Wales \(Representation of the People\) \(Amendment\) Order 2016](#)

**Parameters** **poll\_date** – a datetime representing the date of the poll

**Returns** a datetime representing the expected publish date

**northern\_ireland\_assembly** (*poll\_date: datetime.date*) → datetime.date

Calculate the publish date for an election to the Northern Ireland Assembly

This is set out by Schedule 5, Rules 1 and 2 of [The Northern Ireland Assembly \(Elections\) \(Amendment\) Order 2009](#)

**Parameters** **poll\_date** – a datetime representing the date of the poll

**Returns** a datetime representing the expected publish date

**police\_and\_crime\_commissioner** (*poll\_date: datetime.date*) → datetime.date

Calculate the publish date for an election to the position of Police and Crime Commissioner

This is set out in [The Police and Crime Commissioner Elections \(Amendment\) Order 2014](#)

**Parameters** **poll\_date** – a datetime representing the date of the poll

**Returns** a datetime representing the expected publish date

**scottish\_parliament** (*poll\_date: datetime.date*) → datetime.date

Calculate the publish date for an election to the Scottish Parliament

This is set out in [The Scottish Parliament \(Elections etc.\) Order 2015](#)

**Parameters** `poll_date` – a datetime representing the date of the poll

**Returns** a datetime representing the expected publish date

**uk\_parliament** (*poll\_date: datetime.date, country: sopn\_publish\_date.calendars.Country = None*)

Calculate the publish date for an election to the Parliament of the United Kingdom

This is set out in [Representation of the People Act 1983](#) and its amendments.

**Parameters**

- `poll_date` – a datetime representing the date of the poll
- `country` – an optional Country representing the country where the election will be held

**Returns** a datetime representing the expected publish date

## 4.2 sopn\_publish\_date.calendars

**class** `sopn_publish_date.calendars.BankHolidayCalendar` (*dates*)

A calendar that honours the standard 5-day week in addition to the input list of dates.

**static** `create_matcher_from_entry` (*entry: dict*) → `sopn_publish_date.date.DateMatcher`

`exempted_dates` ()

**class** `sopn_publish_date.calendars.Country`

The countries of the United Kingdom.

`ENGLAND = 1`

`NORTHERN_IRELAND = 2`

`SCOTLAND = 3`

`WALES = 4`

**class** `sopn_publish_date.calendars.FixedDates`

A set of fixed dates used in other calculations by this library.

`EUROPARL_GIBRALTAR_2019 = datetime.date(2019, 4, 24)`

**class** `sopn_publish_date.calendars.Region`

The regions of the United Kingdom and Gibraltar as elected in the EU Parliament

`EAST_MIDLANDS = 1`

`EAST_OF_ENGLAND = 2`

`LONDON = 3`

`NORTHERN_IRELAND = 6`

`NORTH_EAST_ENGLAND = 4`

`NORTH_WEST_ENGLAND = 5`

`SCOTLAND = 7`

`SOUTH_EAST_ENGLAND = 8`

```
SOUTH_WEST_ENGLAND = 9
```

```
WALES = 10
```

```
WEST_MIDLANDS = 11
```

```
YORKSHIRE_AND_THE_HUMBER = 12
```

```
class sopn_publish_date.calendars.UnitedKingdomBankHolidays
```

A representation of the bank holiday calendars in the United Kingdom.

This class exposes a function for each unique calendar: England & Wales, Northern Ireland, and Scotland.

```
england_and_wales () → sopn_publish_date.calendars.BankHolidayCalendar
```

**Returns** a calendar representation of bank holidays in England and Wales

```
from_country (country: sopn_publish_date.calendars.Country) →  
sopn_publish_date.calendars.BankHolidayCalendar
```

Return the bank holiday calendar for the input country.

**Parameters** **country** – the country to retrieve the calendar for

**Returns** the corresponding calendar

```
northern_ireland () → sopn_publish_date.calendars.BankHolidayCalendar
```

**Returns** a calendar representation of bank holidays in Northern Ireland

```
scotland () → sopn_publish_date.calendars.BankHolidayCalendar
```

**Returns** a calendar representation of bank holidays in Scotland

```
sopn_publish_date.calendars.working_days_before (end_date: datetime.date,  
days: int, calendar: sopn_publish_date.calendars.BankHolidayCalendar)  
→ datetime.date
```

Return date corresponding to *count* working days before *poll\_date* according to the given bank holiday calendar

**Parameters**

- **end\_date** – the date of the poll
- **days** – the number of days before the poll date
- **calendar** – the bank holiday calendar used in the calculation

**Returns** the calculated date

## 4.3 sopn\_publish\_date.election\_ids

```
exception sopn_publish_date.election_ids.AmbiguousElectionIdError (election_id:  
str)
```

An exception type to represent when an election id (usually a group such as *local.2019-05-02*) can correspond to elections in multiple countries with different legislation governing the publish date of Statements of Persons Nominated.

```
exception sopn_publish_date.election_ids.InvalidElectionIdError (election_id:  
str)
```

An exception type to represent when an election id does not confirm to DemocracyClub's *uk-election-ids* format

```
exception sopn_publish_date.election_ids.NoSuchElectionTypeError (election_type:  
str)
```

An exception type to represent when an election type doesn't actually represent a valid election.

`sopn_publish_date.election_ids.type_and_poll_date` (*election\_id: str*) -> (<class 'str'>, <class 'datetime.date'>)

Extract `election_type` (e.g. `parl`, `local`, `mayor`) and `poll_date` from an election id.

**Parameters** `election_id` – a string representing an election id in `uk-election-ids` format

**Returns** a tuple (`election_type: str`, `poll_date: date`)

## 4.4 sopn\_publish\_date.date

**class** `sopn_publish_date.date.DateMatcher` (*day: int, month: int, year: int = None, name: str = None*)

An object that represents a matcher against `datetime.date` objects, given a day, a month, and an optional year.

**matches** (*other: datetime.date*) → bool

Return whether the input date matches the attributes of this class

**Parameters** `other` – the date being matched against

:return a boolean representing whether the input matched against this class's attributes

`sopn_publish_date.date.days_before` (*poll\_date: datetime.date, days: int, ignore: List[sopn\_publish\_date.date.DateMatcher] = None*) → `datetime.date`

Return date corresponding to `days` working days before `poll_date`, not counting the list of provided exemptions

**Parameters**

- `poll_date` – the date of the poll
- `days` – the number of days before the poll date
- `ignore` – the list of `DateMatchers` to ignore in the look-back calculation

**Returns** the calculated date





**S**

`sopn_publish_date`, [7](#)  
`sopn_publish_date.calendars`, [9](#)  
`sopn_publish_date.date`, [11](#)  
`sopn_publish_date.election_ids`, [10](#)



## A

AmbiguousElectionIdError, 10

## B

BankHolidayCalendar (class in *sopn\_publish\_date.calendars*), 9

## C

Country (class in *sopn\_publish\_date.calendars*), 9  
 create\_matcher\_from\_entry() (sopn\_publish\_date.calendars.BankHolidayCalendar static method), 9

## D

DateMatcher (class in *sopn\_publish\_date.date*), 11  
 days\_before() (in module *sopn\_publish\_date.date*), 11

## E

EAST\_MIDLANDS (*sopn\_publish\_date.calendars.Region* attribute), 9  
 EAST\_OF\_ENGLAND (*sopn\_publish\_date.calendars.Region* attribute), 9  
 ENGLAND (*sopn\_publish\_date.calendars.Country* attribute), 9  
 england\_and\_wales() (*sopn\_publish\_date.calendars.UnitedKingdomBankHolidays* method), 10  
 EUROPARL\_GIBRALTAR\_2019 (*sopn\_publish\_date.calendars.FixedDates* attribute), 9  
 european\_parliament() (*sopn\_publish\_date.StatementPublishDate* method), 7  
 exempted\_dates() (*sopn\_publish\_date.calendars.BankHolidayCalendar* method), 9

## F

FixedDates (class in *sopn\_publish\_date.calendars*), 9

for\_id() (*sopn\_publish\_date.StatementPublishDate* method), 7

from\_country() (*sopn\_publish\_date.calendars.UnitedKingdomBankHolidays* method), 10

## G

greater\_london\_assembly() (*sopn\_publish\_date.StatementPublishDate* method), 7

## I

InvalidElectionIdError, 10

## L

local() (*sopn\_publish\_date.StatementPublishDate* method), 8

LONDON (*sopn\_publish\_date.calendars.Region* attribute), 9

## M

matches() (*sopn\_publish\_date.date.DateMatcher* method), 11

mayor() (*sopn\_publish\_date.StatementPublishDate* method), 8

## N

north\_west\_assembly\_for\_wales() (*sopn\_publish\_date.StatementPublishDate* method), 8

NORTH\_EAST\_ENGLAND (*sopn\_publish\_date.calendars.Region* attribute), 9

NORTH\_WEST\_ENGLAND (*sopn\_publish\_date.calendars.Region* attribute), 9

NORTHERN\_IRELAND (*sopn\_publish\_date.calendars.Country* attribute), 9

NORTHERN\_IRELAND (*sopn\_publish\_date.calendars.Region* attribute), 9

northern\_ireland() (sopn\_publish\_date.calendars.UnitedKingdomBankHolidays method), 10

northern\_ireland\_assembly() (sopn\_publish\_date.StatementPublishDate method), 8

NoSuchElectionTypeError, 10

## P

police\_and\_crime\_commissioner() (sopn\_publish\_date.StatementPublishDate method), 8

## R

Region (class in sopn\_publish\_date.calendars), 9

## S

SCOTLAND (sopn\_publish\_date.calendars.Country attribute), 9

SCOTLAND (sopn\_publish\_date.calendars.Region attribute), 9

scotland() (sopn\_publish\_date.calendars.UnitedKingdomBankHolidays method), 10

scottish\_parliament() (sopn\_publish\_date.StatementPublishDate method), 8

sopn\_publish\_date (module), 7

sopn\_publish\_date.calendars (module), 9

sopn\_publish\_date.date (module), 11

sopn\_publish\_date.election\_ids (module), 10

SOUTH\_EAST\_ENGLAND (sopn\_publish\_date.calendars.Region attribute), 9

SOUTH\_WEST\_ENGLAND (sopn\_publish\_date.calendars.Region attribute), 9

StatementPublishDate (class in sopn\_publish\_date), 7

## T

type\_and\_poll\_date() (in module sopn\_publish\_date.election\_ids), 10

## U

uk\_parliament() (sopn\_publish\_date.StatementPublishDate method), 9

UnitedKingdomBankHolidays (class in sopn\_publish\_date.calendars), 10

## W

WALES (sopn\_publish\_date.calendars.Country attribute), 9

WALES (sopn\_publish\_date.calendars.Region attribute), 10

WEST\_MIDLANDS (sopn\_publish\_date.calendars.Region attribute), 10

working\_days\_before() (in module sopn\_publish\_date.calendars), 10

## Y

YORKSHIRE\_AND\_THE\_HUMBER (sopn\_publish\_date.calendars.Region attribute), 10