

# Package ‘VedicDateTime’

September 20, 2023

**Type** Package

**Title** Vedic Calendar System

**Version** 0.1.9

**Description** Provides platform for Vedic calendar system having several functionalities to facilitate conversion between Gregorian and Vedic calendar systems, and helpful in examining its impact in the time series analysis domain.

**URL** <https://www.neerajbokde.in/vignette/2022-09-05-VedicDateTime>

**BugReports** <https://github.com/prajwalkpatil/VedicDateTime/issues>

**License** GPL (>= 3)

**Encoding** UTF-8

**RoxygenNote** 7.2.3

**Depends** R (>= 3.1.0)

**Suggests** knitr, rmarkdown, testthat (>= 3.0.0), qpdf, formatR, spelling, tinytex

**VignetteBuilder** knitr

**Imports** swephR

**Config/testthat/edition** 3

**Language** en-US

**NeedsCompilation** no

**Author** Neeraj Dhanraj Bokde [aut, cre, cph],  
Prajwal Kailasnath Patil [aut],  
Saradindu Sengupta [aut],  
Andrés Elías Feijóo Lorenzo [aut]

**Maintainer** Neeraj Dhanraj Bokde <neerajdhanraj@gmail.com>

**Repository** CRAN

**Date/Publication** 2023-09-20 05:20:02 UTC

**R topics documented:**

ahargana . . . . .	3
day_duration . . . . .	3
elapsed_year . . . . .	4
from_dms . . . . .	4
get_karana_name . . . . .	5
get_lagna_name . . . . .	5
get_masa_name . . . . .	6
get_nakshatra_name . . . . .	6
get_rashi_name . . . . .	7
get_ritu_name . . . . .	7
get_samvatsara_name . . . . .	8
get_tithi_name . . . . .	8
get_vaara_name . . . . .	9
get_yoga_name . . . . .	9
gregorian_to_jd . . . . .	10
inverse_lagrange . . . . .	10
jd_to_gregorian . . . . .	11
karana . . . . .	11
karanas . . . . .	12
lagna . . . . .	12
lunar_phase . . . . .	13
masa . . . . .	13
masas . . . . .	14
moonrise . . . . .	14
moonset . . . . .	15
moon_longitude . . . . .	15
nakshatra . . . . .	16
nakshatras . . . . .	16
new_moon . . . . .	17
rashi . . . . .	17
rashis . . . . .	18
ritu . . . . .	18
ritus . . . . .	19
samvatsara . . . . .	19
samvatsars . . . . .	20
sunrise . . . . .	20
sunset . . . . .	21
sun_longitude . . . . .	21
tithi . . . . .	22
tithis . . . . .	22
to_dms . . . . .	23
unwrap_angles . . . . .	23
vaara . . . . .	24
vaaras . . . . .	24
yoga . . . . .	25
yogas . . . . .	25

*ahargana*

3

## Index

[26](#)

---

<i>ahargana</i>	<i>ahargana</i>
-----------------	-----------------

---

### Description

*ahargana*

### Usage

`ahargana(jd)`

### Arguments

`jd`                    Julian day number

### Value

Ahargana

### Examples

```
ahargana(2459778)
ahargana(swephR::swe_julday(2022, 7, 14, 0, swephR::SE$GREG_CAL))
```

---

<i>day_duration</i>	<i>day_duration</i>
---------------------	---------------------

---

### Description

Duration of the day for a given place and time

### Usage

`day_duration(jd, place)`

### Arguments

`jd`                    Julian day number  
`place`                Vector containing latitude, longitude and timezone

### Value

Vector containing the length of the day & in dms

### Examples

```
day_duration(2459778, c(15.34, 75.13, +5.5))
day_duration(swephR::swe_julday(2022, 7, 14, 0, swephR::SE$GREG_CAL), c(15.34, 75.13, +5.5))
```

---

elapsed_year	<i>elapsed_year</i>
--------------	---------------------

---

**Description**

elapsed\_year

**Usage**

elapsed\_year(jd, maasa\_num)

**Arguments**

jd	Julian Day number
maasa_num	Number indicating the Maasa

**Value**

A vector containing Kali, Saka, and Vikram Samvat

**Examples**

elapsed\_year(2459778,2)

---

from_dms	<i>from_dms</i>
----------	-----------------

---

**Description**

Convert degrees, minutes, and seconds to decimal degrees

**Usage**

from\_dms(degs, mins, secs)

**Arguments**

degs	Degrees
mins	Minutes
secs	Seconds

**Value**

Degrees as a decimal number

**Examples**

from\_dms(30, 15, 50)

---

*get\_karana\_name*      *get\_karana\_name*

---

**Description**

Get name(s) of the Karana for given Julian day number and place.

**Usage**

`get_karana_name(jd, place)`

**Arguments**

jd                    Julian day number  
place                Vector containing latitude, longitude and timezone

**Value**

Name(s) of the Karana.

**Examples**

```
get_karana_name(2459778,c(15.34, 75.13, +5.5))  
get_karana_name(swephR::swe_julday(2022,7,14,0,swephR::SE$GREG_CAL),c(15.34, 75.13, +5.5))
```

---

*get\_lagna\_name*      *get\_lagna\_name*

---

**Description**

Get name of the Lagna for given Julian day number.

**Usage**

`get_lagna_name(jd)`

**Arguments**

jd                    Julian day number

**Value**

Name of the lagna.

**Examples**

```
get_lagna_name(2459778)  
get_lagna_name(gregorian_to_jd(30,8,2022))
```

---

<code>get_masa_name</code>	<i>get_masa_name</i>
----------------------------	----------------------

---

**Description**

Get name of the Masa for given Julian day number and place.

**Usage**

```
get_masa_name(jd, place)
```

**Arguments**

<code>jd</code>	Julian day number
<code>place</code>	Vector containing latitude, longitude and timezone

**Value**

Name of the Masa

**Examples**

```
get_masa_name(2459778, c(15.34, 75.13, +5.5))
get_masa_name(swephR::swe_julday(2022, 7, 14, 0, swephR::SE$GREG_CAL), c(15.34, 75.13, +5.5))
```

---

<code>get_nakshatra_name</code>	<i>get_nakshatra_name</i>
---------------------------------	---------------------------

---

**Description**

Get name(s) of the Nakshatra for given Julian day number and place.

**Usage**

```
get_nakshatra_name(jd, place)
```

**Arguments**

<code>jd</code>	Julian day number
<code>place</code>	Vector containing latitude, longitude and timezone

**Value**

Name(s) of the Nakshatra and its ending time.

**Examples**

```
get_nakshatra_name(2459778,c(15.34, 75.13, +5.5))  
get_nakshatra_name(swephR::swe_julday(2022,7,14,0,swephR::SE$GREG_CAL),c(15.34, 75.13, +5.5))
```

---

<i>get_rashi_name</i>	<i>get_rashi_name</i>
-----------------------	-----------------------

---

**Description**

Get name of the Rashi for given Julian day number.

**Usage**

```
get_rashi_name(jd)
```

**Arguments**

jd                    Julian day number

**Value**

Name of the Rashi.

**Examples**

```
get_rashi_name(2459778)  
get_rashi_name(gregorian_to_jd(30,8,2022))
```

---

<i>get_ritu_name</i>	<i>get_ritu_name</i>
----------------------	----------------------

---

**Description**

*get\_ritu\_name*

**Usage**

```
get_ritu_name(masa_num)
```

**Arguments**

masa\_num            Number associated with a Masa

**Value**

Ritu's name

**Examples**

```
ritu(2)
```

---

```
get_samvatsara_name    get_samvatsara_name
```

---

**Description**

Name of the Shaka Samvatsar for a given Julian day number and maasa number.

**Usage**

```
get_samvatsara_name(jd, maasa_num)
```

**Arguments**

jd	Julian day number
maasa_num	Maasa number

**Value**

Shaka Samvatsar

**Examples**

```
get_samvatsara_name(2459778,2)
```

---

```
get_tithi_name        get_tithi_name
```

---

**Description**

Get name(s) of the Tithi for given Julian day number and place.

**Usage**

```
get_tithi_name(jd, place)
```

**Arguments**

jd	Julian day number
place	Vector containing latitude, longitude and timezone

**Value**

Name(s) of the Tithi and its ending time.



**Examples**

```
get_tithi_name(2459778,c(15.34, 75.13, +5.5))  
get_tithi_name(swephR::swe_julday(2022,7,14,0,swephR::SE$GREG_CAL),c(15.34, 75.13, +5.5))
```

---

get_vaara_name	<i>get_vaara_name</i>
----------------	-----------------------

---

**Description**

Get name of the Vaara for given Julian day number.

**Usage**

```
get_vaara_name(jd)
```

**Arguments**

jd                    Julian day number

**Value**

Name of the Vaara.

**Examples**

```
get_vaara_name(2459778)  
get_vaara_name(swephR::swe_julday(2022,7,14,0,swephR::SE$GREG_CAL))
```

---

get_yoga_name	<i>get_yoga_name</i>
---------------	----------------------

---

**Description**

Get name(s) of the Yoga for given Julian day number and place.

**Usage**

```
get_yoga_name(jd, place)
```

**Arguments**

jd                    Julian day number  
place                 Vector containing latitude, longitude and timezone

**Value**

Name(s) of the Yoga and its ending time.

**Examples**

```
get_yoga_name(2459778,c(15.34, 75.13, +5.5))
get_yoga_name(swephR::swe_julday(2022,7,14,0,swephR::SE$GREG_CAL),c(15.34, 75.13, +5.5))
```

---

```
gregorian_to_jd      gregorian_to_jd
```

---

**Description**

Convert Gregorian date to Julian day number at 00:00 UTC

**Usage**

```
gregorian_to_jd(day, month, year)
```

**Arguments**

day	Day number
month	Month number
year	Year number

**Value**

Julian day number

**Examples**

```
gregorian_to_jd(18,7,2022)
```

---

```
inverse_lagrange      inverse_lagrange
```

---

**Description**

Given two vectors  $x$  and  $y$ , find the value of  $x = x_a$  when  $y = y_a$ , i.e.,  $f(x_a) = y_a$

**Usage**

```
inverse_lagrange(x, y, ya)
```

**Arguments**

$x$	Vector $x$
$y$	Vector $y$
$y_a$	Double $y_a$

**Value**

Value of xa

---

*jd\_to\_gregorian*      *jd\_to\_gregorian*

---

**Description**

Convert Julian day number to Gregorian date

**Usage**

`jd_to_gregorian(jd)`

**Arguments**

jd                      Julian day number

**Value**

Gregorian date

**Examples**

`jd_to_gregorian(2459778)`

---

*karana*                      *karana*

---

**Description**

Karana for a given place and time

**Usage**

`karana(jd, place)`

**Arguments**

jd                      Julian day number  
place                  Vector containing latitude, longitude and timezone

**Value**

Two karanas

**Examples**

```
karana(2459778,c(15.34, 75.13, +5.5))
karana(gregorian_to_jd(17,6,2022),c(15.34, 75.13, +5.5))
```

---

karanas	<i>karanas</i>
---------	----------------

---

**Description**

Name of 60 Karanas which is when moon traverses 6° in longitude relative to the sun

**Usage**

```
karanas
```

**Format**

An object of class character of length 60.

---

lagna	<i>Lagna</i>
-------	--------------

---

**Description**

Lagna for a given Julian day number

**Usage**

```
lagna(jd)
```

**Arguments**

jd	Julian day number
----	-------------------

**Value**

Lagna as an integer

**Examples**

```
lagna(2459778)
lagna(gregorian_to_jd(30,8,2022))
```

---

lunar_phase	<i>lunar_phase</i>
-------------	--------------------

---

**Description**

Lunar phase for a given Julian day number

**Usage**

```
lunar_phase(jd)
```

**Arguments**

jd	Julian day number
----	-------------------

**Value**

Lunar phase

**Examples**

```
lunar_phase(2459778)
```

---

masa	<i>masa</i>
------	-------------

---

**Description**

Masa for a given place and time

**Usage**

```
masa(jd, place)
```

**Arguments**

jd	Julian day number
place	Vector containing latitude, longitude and timezone

**Value**

Masa number and whether it is adhika masa or not

**Examples**

```
masa(2459778, c(15.34, 75.13, +5.5))
masa(swephR::swe_julday(2022, 7, 14, 0, swephR::SE$GREG_CAL), c(15.34, 75.13, +5.5))
```

---

masas	<i>masas</i>
-------	--------------

---

**Description**

Lunar month in the Vedic calendar system

**Usage**

masas

**Format**

An object of class character of length 12.

---

moonrise	<i>moonrise</i>
----------	-----------------

---

**Description**

Moonrise for a given date and place

**Usage**

moonrise(jd, place)

**Arguments**

jd	Julian day number
place	Vector containing latitude, longitude and timezone

**Value**

Moonrise as Julian day number

**Examples**

moonrise(2459778,c(15.34, 75.13, +5.5))

---

moonset	<i>moonset</i>
---------	----------------

---

**Description**

Moonset for a given date and place

**Usage**

```
moonset(jd, place)
```

**Arguments**

jd	Julian day number
place	Vector containing latitude, longitude and timezone

**Value**

Moonset as Julian day number

**Examples**

```
moonset(2459778,c(15.34, 75.13, +5.5))
```

---

moon_longitude	<i>moon_longitude</i>
----------------	-----------------------

---

**Description**

Get Lunar longitude for a given Julian day number.

**Usage**

```
moon_longitude(jd)
```

**Arguments**

jd	Julian day
----	------------

**Value**

Lunar longitude for jd

**Examples**

```
moon_longitude(2459778)  
moon_longitude(2459500)
```

---

nakshatra	<i>nakshatra</i>
-----------	------------------

---

**Description**

Nakshatra for a given place and time

**Usage**

nakshatra(jd, place)

**Arguments**

jd	Julian day number
place	Vector containing latitude, longitude and timezone

**Value**

Nakshatra and it's ending time

**Examples**

```
nakshatra(2459778,c(15.34, 75.13, +5.5))
nakshatra(gregorian_to_jd(17,6,2022),c(15.34, 75.13, +5.5))
```

---

nakshatras	<i>nakshatras</i>
------------	-------------------

---

**Description**

Name of the 27 Nakshatras in Vedic calendar system

**Usage**

nakshatras

**Format**

An object of class character of length 27.



---

new_moon	<i>new_moon</i>
----------	-----------------

---

**Description**

Julian day representing the new moon day for a given Julian day number and tithi

**Usage**

```
new_moon(jd, tithi_, opt = -1)
```

**Arguments**

jd	Julian day number
tithi_	Number associated with the tithi
opt	Option to select next new moon day(opt = 1) or previous new moon day (opt = -1), Default opt = -1 .

**Value**

New moon day as a Julian day number

**Examples**

```
new_moon(2459778,2)
new_moon(2459778,tithi(2459778,c(15.34, 75.13, +5.5)))
```

---

rashi	<i>Rashi</i>
-------	--------------

---

**Description**

Rashi for a given Julian day number

**Usage**

```
rashi(jd)
```

**Arguments**

jd	Julian day number
----	-------------------

**Value**

Rashi as an integer

**Examples**

```
rashi(2459778)
rashi(gregorian_to_jd(30, 8, 2022))
```

---

rashis	<i>rashis</i>
--------	---------------

---

**Description**

The name of 12 Rashis which represents the position of the moon on the zodiac at a given time

**Usage**

```
rashis
```

**Format**

An object of class character of length 12.

---

ritu	<i>ritu</i>
------	-------------

---

**Description**

```
ritu
```

**Usage**

```
ritu(masa_num)
```

**Arguments**

masa_num	Number associated with a Masa
----------	-------------------------------

**Value**

Number associated with the Ritu

**Examples**

```
ritu(2)
```

---

ritus

*ritus*

---

**Description**

Name of the 6 seasons in Vedic calendar system

**Usage**

ritus

**Format**

An object of class character of length 6.

---

samvatsara

*samvatsara*

---

**Description**

Shaka Samvatsar for a given Julian day number and maasa number.

**Usage**

samvatsara(jd, maasa\_num)

**Arguments**

jd                    Julian day number

maasa\_num          Maasa number

**Value**

Number associated with the Shaka Samvatsar

**Examples**

samvatsara(2459778, 2)

samvatsars

*samvatsars*

---

**Description**

Name of the Year in Hindu Panchang

**Usage**

samvatsars

**Format**

An object of class character of length 60.

---

sunrise

*sunrise*

---

**Description**

Sunrise for a given date and place

**Usage**

sunrise(jd, place)

**Arguments**

jd                    Julian day number

place                Vector containing latitude, longitude and timezone

**Value**

Sunrise as Julian day number

**Examples**

sunrise(2459778,c(15.34, 75.13, +5.5))

---

sunset	<i>sunset</i>
--------	---------------

---

**Description**

Sunset for a given date and place

**Usage**

```
sunset(jd, place)
```

**Arguments**

jd	Julian day number
place	Vector containing latitude, longitude and timezone

**Value**

Sunset as Julian day number

**Examples**

```
sunset(2459778, c(15.34, 75.13, +5.5))
```

---

sun_longitude	<i>sun_longitude</i>
---------------	----------------------

---

**Description**

Get Solar longitude for a given Julian day number.

**Usage**

```
sun_longitude(jd)
```

**Arguments**

jd	Julian day
----	------------

**Value**

Solar longitude for jd

**Examples**

```
sun_longitude(2459778)  
sun_longitude(2459500)
```

---

tithi	<i>tithi</i>
-------	--------------

---

**Description**

Tithi for a given place and time

**Usage**

```
tithi(jd, place)
```

**Arguments**

jd	Julian day number
place	Vector containing latitude, longitude and timezone

**Value**

Tithi and its ending time

**Examples**

```
tithi(2459778,c(15.34, 75.13, +5.5))
tithi(gregorian_to_jd(17,6,2022),c(15.34, 75.13, +5.5))
```

---

tithis	<i>tithis</i>
--------	---------------

---

**Description**

lunar day in the Vedic calendar system

**Usage**

```
tithis
```

**Format**

An object of class character of length 30.

---

to_dms	<i>to_dms</i>
--------	---------------

---

**Description**

Convert decimal degrees to degrees, minutes, and seconds

**Usage**

```
to_dms(deg)
```

**Arguments**

deg                    Degrees as a decimal number

**Value**

A vector containing degrees, minutes and seconds

**Examples**

```
to_dms(30.263888889)
```

---

unwrap_angles	<i>unwrap_angles</i>
---------------	----------------------

---

**Description**

Add 360 degs an element in the input vector if elements are not sorted in ascending order.

**Usage**

```
unwrap_angles(angles)
```

**Arguments**

angles                Vector containing angles

**Value**

angles in ascending order

---

vaara

*vaara*

---

### Description

Vaara for a given Julian day number

### Usage

vaara(jd)

### Arguments

jd                    Julian day number

### Value

Vaara as an integer

### Examples

vaara(2459778)

---

vaaras

*vaaras*

---

### Description

Name of the day of the week

### Usage

vaaras

### Format

An object of class character of length 7.



---

yoga	<i>yoga</i>
------	-------------

---

**Description**

Yoga for a given place and time

**Usage**

```
yoga(jd, place)
```

**Arguments**

jd	Julian day number
place	Vector containing latitude, longitude and timezone

**Value**

Yoga and it's ending time

**Examples**

```
yoga(2459778,c(15.34, 75.13, +5.5))
yoga(gregorian_to_jd(17,6,2022),c(15.34, 75.13, +5.5))
```

---

yogas	<i>yogas</i>
-------	--------------

---

**Description**

Name of the 27 yogas which is sum of sidereal longitudes of sun and moon in the multiples of 13 degrees 20 minutes

**Usage**

```
yogas
```

**Format**

An object of class character of length 27.

# Index

## \* datasets

- [karanas, 12](#)
  - [masas, 14](#)
  - [nakshatras, 16](#)
  - [rashis, 18](#)
  - [ritus, 19](#)
  - [samvatsars, 20](#)
  - [tithis, 22](#)
  - [vaaras, 24](#)
  - [yogas, 25](#)
- [ahargana, 3](#)
- [day\\_duration, 3](#)
- [elapsed\\_year, 4](#)
- [from\\_dms, 4](#)
- [get\\_karana\\_name, 5](#)
- [get\\_lagna\\_name, 5](#)
- [get\\_masa\\_name, 6](#)
- [get\\_nakshatra\\_name, 6](#)
- [get\\_rashi\\_name, 7](#)
- [get\\_ritu\\_name, 7](#)
- [get\\_samvatsara\\_name, 8](#)
- [get\\_tithi\\_name, 8](#)
- [get\\_vaara\\_name, 9](#)
- [get\\_yoga\\_name, 9](#)
- [gregorian\\_to\\_jd, 10](#)
- [inverse\\_lagrange, 10](#)
- [jd\\_to\\_gregorian, 11](#)
- [karana, 11](#)
- [karanas, 12](#)
- [lagna, 12](#)
- [lunar\\_phase, 13](#)
- [masa, 13](#)
- [masas, 14](#)
- [moon\\_longitude, 15](#)
- [moonrise, 14](#)
- [moonset, 15](#)
- [nakshatra, 16](#)
- [nakshatras, 16](#)
- [new\\_moon, 17](#)
- [rashi, 17](#)
- [rashis, 18](#)
- [ritu, 18](#)
- [ritus, 19](#)
- [samvatsara, 19](#)
- [samvatsars, 20](#)
- [sun\\_longitude, 21](#)
- [sunrise, 20](#)
- [sunset, 21](#)
- [tithi, 22](#)
- [tithis, 22](#)
- [to\\_dms, 23](#)
- [unwrap\\_angles, 23](#)
- [vaara, 24](#)
- [vaaras, 24](#)
- [yoga, 25](#)
- [yogas, 25](#)