pandas.tseries.offsets.Minute.base

property Minute.**base**
Returns a copy of the calling offset object with n=1 and all other attributes equal.

delta	
freqstr	
kwds	
name	
nanos	
rule_code	

Methods

apply(self, other)	
apply_index(self, other)	Vectorized apply of DateOffset to DatetimeIndex, raises NotImplentedError for offsets without a vectorized implementation.
rollback(self, dt)	Roll provided date backward to next offset only if not on offset.
rollforward(self, dt)	Roll provided date forward to next offset only if not on offset.

pandas.tseries.offsets.Minute.apply

Minute.**apply**(*self*, *other*)

pandas.tseries.offsets.Minute.apply_index

Minute.**apply_index**(*self*, *other*)
Vectorized apply of DateOffset to DatetimeIndex, raises NotImplentedError for offsets without a vectorized implementation.

Parameters

i [DatetimeIndex]

Returns

y [DatetimeIndex]

pandas.tseries.offsets.Minute.rollback

Minute.**rollback**(*self*, *dt*)

Roll provided date backward to next offset only if not on offset.

Returns

TimeStamp Rolled timestamp if not on offset, otherwise unchanged timestamp.

pandas.tseries.offsets.Minute.rollforward

Minute.**rollforward**(*self*, *dt*)

Roll provided date forward to next offset only if not on offset.

Returns

TimeStamp Rolled timestamp if not on offset, otherwise unchanged timestamp.

__call__	
copy	
isAnchored	
is_anchored	
is_on_offset	
onOffset	

Properties

Minute.delta
Minute.freqstr
Minute.kwds
Minute.name
Minute.nanos
Minute.normalize
Minute.rule_code

pandas.tseries.offsets.Minute.delta

property Minute.**delta**

pandas.tseries.offsets.Minute.freqstr

Minute.**freqstr**

pandas.tseries.offsets.Minute.kwds

property Minute.**kwds**

pandas.tseries.offsets.Minute.name

property Minute.**name**

pandas.tseries.offsets.Minute.nanos

property Minute.**nanos**

pandas.tseries.offsets.Minute.normalize

Minute.**normalize = False**

pandas.tseries.offsets.Minute.rule_code

property Minute.**rule_code**

Methods

Minute.copy(self)	
Minute.isAnchored(self)	
Minute.onOffset(self, dt)	
Minute.is_anchored(self)	
Minute.is_on_offset(self, dt)	
Minute.__call__(self, other)	Call self as a function.

pandas.tseries.offsets.Minute.copy

Minute.**copy**(*self*)

pandas.tseries.offsets.Minute.isAnchored

Minute.**isAnchored**(*self*)

pandas.tseries.offsets.Minute.onOffset

Minute.**onOffset**(*self*, *dt*)

pandas.tseries.offsets.Minute.is_anchored

Minute.**is_anchored**(*self*)

pandas.tseries.offsets.Minute.is_on_offset

Minute.**is_on_offset**(*self*, *dt*)

pandas.tseries.offsets.Minute.__call__

Minute.**__call__**(*self*, *other*)
Call self as a function.

3.8.36 Second

Second([n, normalize])	

Attributes

pandas.tseries.offsets.Second

class pandas.tseries.offsets.**Second**(*n=1*, *normalize=False*)

Attributes

base	Returns a copy of the calling offset object with n=1 and all other attributes equal.

pandas.tseries.offsets.Second.base

property Second.**base**
Returns a copy of the calling offset object with n=1 and all other attributes equal.

delta	
freqstr	
kwds	
name	
nanos	
rule_code	

Methods

apply(self, other)	
apply_index(self, other)	Vectorized apply of DateOffset to DatetimeIndex, raises NotImplentedError for offsets without a vectorized implementation.
rollback(self, dt)	Roll provided date backward to next offset only if not on offset.
rollforward(self, dt)	Roll provided date forward to next offset only if not on offset.

pandas.tseries.offsets.Second.apply

Second.**apply**(*self*, *other*)

pandas.tseries.offsets.Second.apply_index

Second.**apply_index**(*self*, *other*)
Vectorized apply of DateOffset to DatetimeIndex, raises NotImplentedError for offsets without a vectorized implementation.

Parameters

i [DatetimeIndex]

Returns

y [DatetimeIndex]

pandas.tseries.offsets.Second.rollback

Second.**rollback**(*self*, *dt*)

Roll provided date backward to next offset only if not on offset.

Returns

TimeStamp Rolled timestamp if not on offset, otherwise unchanged timestamp.

pandas.tseries.offsets.Second.rollforward

Second.**rollforward**(*self*, *dt*)

Roll provided date forward to next offset only if not on offset.

Returns

TimeStamp Rolled timestamp if not on offset, otherwise unchanged timestamp.

__call__	
copy	
isAnchored	
is_anchored	
is_on_offset	
onOffset	

Properties

Second.delta
Second.freqstr
Second.kwds
Second.name
Second.nanos
Second.normalize
Second.rule_code

pandas.tseries.offsets.Second.delta

property Second.**delta**

pandas.tseries.offsets.Second.freqstr

Second.**freqstr**

pandas.tseries.offsets.Second.kwds

property Second.**kwds**

pandas.tseries.offsets.Second.name

property Second.**name**

pandas.tseries.offsets.Second.nanos

property Second.**nanos**

pandas.tseries.offsets.Second.normalize

Second.**normalize = False**

pandas.tseries.offsets.Second.rule_code

property Second.**rule_code**

Methods

Second.copy(self)	
Second.isAnchored(self)	
Second.onOffset(self, dt)	
Second.is_anchored(self)	
Second.is_on_offset(self, dt)	
Second.__call__(self, other)	Call self as a function.

pandas.tseries.offsets.Second.copy

Second.**copy**(*self*)

pandas.tseries.offsets.Second.isAnchored

Second.**isAnchored**(*self*)

pandas.tseries.offsets.Second.onOffset

Second.**onOffset**(*self*, *dt*)

pandas.tseries.offsets.Second.is_anchored

Second.**is_anchored**(*self*)

pandas.tseries.offsets.Second.is_on_offset

Second.**is_on_offset**(*self*, *dt*)

pandas.tseries.offsets.Second.__call__

Second.**__call__**(*self*, *other*)
Call self as a function.

3.8.37 Milli

Milli([n, normalize])	

Attributes

pandas.tseries.offsets.Milli

class pandas.tseries.offsets.**Milli**(*n=1*, *normalize=False*)

Attributes

base	Returns a copy of the calling offset object with n=1 and all other attributes equal.

pandas.tseries.offsets.Milli.base

property Milli.**base**
Returns a copy of the calling offset object with n=1 and all other attributes equal.

delta	
freqstr	
kwds	
name	
nanos	
rule_code	

Methods

apply(self, other)	
apply_index(self, other)	Vectorized apply of DateOffset to DatetimeIndex, raises NotImplentedError for offsets without a vectorized implementation.
rollback(self, dt)	Roll provided date backward to next offset only if not on offset.
rollforward(self, dt)	Roll provided date forward to next offset only if not on offset.

pandas.tseries.offsets.Milli.apply

Milli.**apply**(*self*, *other*)

pandas.tseries.offsets.Milli.apply_index

Milli.**apply_index**(*self*, *other*)
Vectorized apply of DateOffset to DatetimeIndex, raises NotImplentedError for offsets without a vectorized implementation.

Parameters

i [DatetimeIndex]

Returns

y [DatetimeIndex]

pandas.tseries.offsets.Milli.rollback

Milli.**rollback**(*self*, *dt*)

Roll provided date backward to next offset only if not on offset.

Returns

TimeStamp Rolled timestamp if not on offset, otherwise unchanged timestamp.

pandas.tseries.offsets.Milli.rollforward

Milli.**rollforward**(*self*, *dt*)

Roll provided date forward to next offset only if not on offset.

Returns

TimeStamp Rolled timestamp if not on offset, otherwise unchanged timestamp.

__call__	
copy	
isAnchored	
is_anchored	
is_on_offset	
onOffset	

Properties

Milli.delta
Milli.freqstr
Milli.kwds
Milli.name
Milli.nanos
Milli.normalize
Milli.rule_code

pandas.tseries.offsets.Milli.delta

property Milli.**delta**

pandas.tseries.offsets.Milli.freqstr

Milli.**freqstr**

pandas.tseries.offsets.Milli.kwds

property Milli.**kwds**

pandas.tseries.offsets.Milli.name

property Milli.**name**

pandas.tseries.offsets.Milli.nanos

property Milli.**nanos**

pandas.tseries.offsets.Milli.normalize

Milli.**normalize = False**

pandas.tseries.offsets.Milli.rule_code

property Milli.**rule_code**

Methods

Milli.copy(self)	
Milli.isAnchored(self)	
Milli.onOffset(self, dt)	
Milli.is_anchored(self)	
Milli.is_on_offset(self, dt)	
Milli.__call__(self, other)	Call self as a function.

pandas.tseries.offsets.Milli.copy

Milli.**copy**(*self*)

pandas.tseries.offsets.Milli.isAnchored

Milli.**isAnchored**(*self*)

pandas.tseries.offsets.Milli.onOffset

Milli.**onOffset**(*self*, *dt*)

pandas.tseries.offsets.Milli.is_anchored

Milli.**is_anchored**(*self*)

pandas.tseries.offsets.Milli.is_on_offset

Milli.**is_on_offset**(*self*, *dt*)

pandas.tseries.offsets.Milli.__call__

Milli.**__call__**(*self*, *other*)
Call self as a function.

3.8.38 Micro

Micro([n, normalize])	

Attributes

pandas.tseries.offsets.Micro

class pandas.tseries.offsets.**Micro**(*n=1*, *normalize=False*)

Attributes

base	Returns a copy of the calling offset object with n=1 and all other attributes equal.

pandas.tseries.offsets.Micro.base

property Micro.**base**
Returns a copy of the calling offset object with n=1 and all other attributes equal.

delta	
freqstr	
kwds	
name	
nanos	
rule_code	

Methods

apply(self, other)	
apply_index(self, other)	Vectorized apply of DateOffset to DatetimeIndex, raises NotImplentedError for offsets without a vectorized implementation.
rollback(self, dt)	Roll provided date backward to next offset only if not on offset.
rollforward(self, dt)	Roll provided date forward to next offset only if not on offset.

pandas.tseries.offsets.Micro.apply

Micro.**apply**(*self*, *other*)

pandas.tseries.offsets.Micro.apply_index

Micro.**apply_index**(*self*, *other*)
Vectorized apply of DateOffset to DatetimeIndex, raises NotImplentedError for offsets without a vectorized implementation.

Parameters

i [DatetimeIndex]

Returns

y [DatetimeIndex]

pandas.tseries.offsets.Micro.rollback

Micro.**rollback**(*self*, *dt*)

Roll provided date backward to next offset only if not on offset.

Returns

TimeStamp Rolled timestamp if not on offset, otherwise unchanged timestamp.

pandas.tseries.offsets.Micro.rollforward

Micro.**rollforward**(*self*, *dt*)

Roll provided date forward to next offset only if not on offset.

Returns

TimeStamp Rolled timestamp if not on offset, otherwise unchanged timestamp.

__call__	
copy	
isAnchored	
is_anchored	
is_on_offset	
onOffset	

Properties

Micro.delta
Micro.freqstr
Micro.kwds
Micro.name
Micro.nanos
Micro.normalize
Micro.rule_code

pandas.tseries.offsets.Micro.delta

property Micro.**delta**

pandas.tseries.offsets.Micro.freqstr

Micro.**freqstr**

pandas.tseries.offsets.Micro.kwds

property Micro.**kwds**

pandas.tseries.offsets.Micro.name

property Micro.**name**

pandas.tseries.offsets.Micro.nanos

property Micro.**nanos**

pandas.tseries.offsets.Micro.normalize

Micro.**normalize = False**

pandas.tseries.offsets.Micro.rule_code

property Micro.**rule_code**

Methods

Micro.copy(self)	
Micro.isAnchored(self)	
Micro.onOffset(self, dt)	
Micro.is_anchored(self)	
Micro.is_on_offset(self, dt)	
Micro.__call__(self, other)	Call self as a function.

pandas.tseries.offsets.Micro.copy

Micro.**copy**(*self*)

pandas.tseries.offsets.Micro.isAnchored

Micro.**isAnchored**(*self*)

pandas.tseries.offsets.Micro.onOffset

Micro.**onOffset**(*self*, *dt*)

pandas.tseries.offsets.Micro.is_anchored

Micro.**is_anchored**(*self*)

pandas.tseries.offsets.Micro.is_on_offset

Micro.**is_on_offset**(*self*, *dt*)

pandas.tseries.offsets.Micro.__call__

Micro.**__call__**(*self*, *other*)
Call self as a function.

3.8.39 Nano

Nano([n, normalize])	

Attributes

pandas.tseries.offsets.Nano

class pandas.tseries.offsets.**Nano**(*n=1*, *normalize=False*)

Attributes

base	Returns a copy of the calling offset object with n=1 and all other attributes equal.

pandas.tseries.offsets.Nano.base

property Nano.**base**
Returns a copy of the calling offset object with n=1 and all other attributes equal.

delta	
freqstr	
kwds	
name	
nanos	
rule_code	

Methods

apply(self, other)	
apply_index(self, other)	Vectorized apply of DateOffset to DatetimeIndex, raises NotImplentedError for offsets without a vectorized implementation.
rollback(self, dt)	Roll provided date backward to next offset only if not on offset.
rollforward(self, dt)	Roll provided date forward to next offset only if not on offset.

pandas.tseries.offsets.Nano.apply

Nano.**apply**(*self*, *other*)

pandas.tseries.offsets.Nano.apply_index

Nano.**apply_index**(*self*, *other*)
Vectorized apply of DateOffset to DatetimeIndex, raises NotImplentedError for offsets without a vectorized implementation.

Parameters

i [DatetimeIndex]

Returns

y [DatetimeIndex]

pandas.tseries.offsets.Nano.rollback

Nano.**rollback**(*self*, *dt*)

Roll provided date backward to next offset only if not on offset.

Returns

TimeStamp Rolled timestamp if not on offset, otherwise unchanged timestamp.

pandas.tseries.offsets.Nano.rollforward

Nano.**rollforward**(*self*, *dt*)

Roll provided date forward to next offset only if not on offset.

Returns

TimeStamp Rolled timestamp if not on offset, otherwise unchanged timestamp.

__call__	
copy	
isAnchored	
is_anchored	
is_on_offset	
onOffset	

Properties

Nano.delta
Nano.freqstr
Nano.kwds
Nano.name
Nano.nanos
Nano.normalize
Nano.rule_code

pandas.tseries.offsets.Nano.delta

property Nano.**delta**

pandas.tseries.offsets.Nano.freqstr

Nano.**freqstr**

pandas.tseries.offsets.Nano.kwds

property Nano.**kwds**

pandas.tseries.offsets.Nano.name

property Nano.**name**

pandas.tseries.offsets.Nano.nanos

property Nano.**nanos**

pandas.tseries.offsets.Nano.normalize

Nano.**normalize = False**

pandas.tseries.offsets.Nano.rule_code

property Nano.**rule_code**

Methods

Nano.copy(self)	
Nano.isAnchored(self)	
Nano.onOffset(self, dt)	
Nano.is_anchored(self)	
Nano.is_on_offset(self, dt)	
Nano.__call__(self, other)	Call self as a function.

pandas.tseries.offsets.Nano.copy

Nano.**copy**(*self*)

pandas.tseries.offsets.Nano.isAnchored

Nano.**isAnchored**(*self*)

pandas.tseries.offsets.Nano.onOffset

Nano.**onOffset**(*self*, *dt*)

pandas.tseries.offsets.Nano.is_anchored

Nano.**is_anchored**(*self*)

pandas.tseries.offsets.Nano.is_on_offset

Nano.**is_on_offset**(*self*, *dt*)

pandas.tseries.offsets.Nano.__call__

Nano.**__call__**(*self*, *other*)
Call self as a function.

3.8.40 BDay

BDay	alias of *pandas.tseries.offsets.BusinessDay*

pandas.tseries.offsets.BDay

pandas.tseries.offsets.**BDay**
alias of *pandas.tseries.offsets.BusinessDay*

Properties

BDay.base	Returns a copy of the calling offset object with n=1 and all other attributes equal.
BDay.freqstr	
BDay.kwds	
BDay.name	
BDay.nanos	
BDay.normalize	
BDay.offset	Alias for self._offset.
BDay.rule_code	

pandas.tseries.offsets.BDay.base

property BDay.**base**
Returns a copy of the calling offset object with n=1 and all other attributes equal.

pandas.tseries.offsets.BDay.freqstr

BDay.**freqstr**

pandas.tseries.offsets.BDay.kwds

property BDay.**kwds**

pandas.tseries.offsets.BDay.name

property BDay.**name**

pandas.tseries.offsets.BDay.nanos

property BDay.**nanos**

pandas.tseries.offsets.BDay.normalize

BDay.**normalize = False**

pandas.tseries.offsets.BDay.offset

property BDay.**offset**
Alias for self._offset.

pandas.tseries.offsets.BDay.rule_code

property BDay.**rule_code**

Methods

BDay.apply(self, other)	
BDay.apply_index(self, other)	Vectorized apply of DateOffset to DatetimeIndex, raises NotImplentedError for offsets without a vectorized implementation.
BDay.copy(self)	
BDay.isAnchored(self)	
BDay.onOffset(self, dt)	
BDay.is_anchored(self)	

continues on next page

Table 377 – continued from previous page

BDay.is_on_offset(self, dt)	
BDay.rollback(self, dt)	Roll provided date backward to next offset only if not on offset.
BDay.rollforward(self, dt)	Roll provided date forward to next offset only if not on offset.
BDay.__call__(self, other)	Call self as a function.

pandas.tseries.offsets.BDay.apply

BDay.**apply**(*self*, *other*)

pandas.tseries.offsets.BDay.apply_index

BDay.**apply_index**(*self*, *other*)

Vectorized apply of DateOffset to DatetimeIndex, raises NotImplentedError for offsets without a vectorized implementation.

Parameters

i [DatetimeIndex]

Returns

y [DatetimeIndex]

pandas.tseries.offsets.BDay.copy

BDay.**copy**(*self*)

pandas.tseries.offsets.BDay.isAnchored

BDay.**isAnchored**(*self*)

pandas.tseries.offsets.BDay.onOffset

BDay.**onOffset**(*self*, *dt*)

pandas.tseries.offsets.BDay.is_anchored

BDay.**is_anchored**(*self*)

pandas.tseries.offsets.BDay.is_on_offset

BDay.**is_on_offset**(*self*, *dt*)

pandas.tseries.offsets.BDay.rollback

BDay.**rollback**(*self*, *dt*)

Roll provided date backward to next offset only if not on offset.

Returns

TimeStamp Rolled timestamp if not on offset, otherwise unchanged timestamp.

pandas.tseries.offsets.BDay.rollforward

BDay.**rollforward**(*self*, *dt*)

Roll provided date forward to next offset only if not on offset.

Returns

TimeStamp Rolled timestamp if not on offset, otherwise unchanged timestamp.

pandas.tseries.offsets.BDay.__call__

BDay.**__call__**(*self*, *other*)

Call self as a function.

3.8.41 BMonthEnd

BMonthEnd	alias of *pandas.tseries.offsets.BusinessMonthEnd*

pandas.tseries.offsets.BMonthEnd

pandas.tseries.offsets.**BMonthEnd**

alias of *pandas.tseries.offsets.BusinessMonthEnd*

Properties

BMonthEnd.base	Returns a copy of the calling offset object with n=1 and all other attributes equal.
BMonthEnd.freqstr	
BMonthEnd.kwds	
BMonthEnd.name	
BMonthEnd.nanos	
BMonthEnd.normalize	
BMonthEnd.rule_code	

pandas.tseries.offsets.BMonthEnd.base

property BMonthEnd.**base**
Returns a copy of the calling offset object with n=1 and all other attributes equal.

pandas.tseries.offsets.BMonthEnd.freqstr

BMonthEnd.**freqstr**

pandas.tseries.offsets.BMonthEnd.kwds

property BMonthEnd.**kwds**

pandas.tseries.offsets.BMonthEnd.name

property BMonthEnd.**name**

pandas.tseries.offsets.BMonthEnd.nanos

property BMonthEnd.**nanos**

pandas.tseries.offsets.BMonthEnd.normalize

BMonthEnd.**normalize = False**

pandas.tseries.offsets.BMonthEnd.rule_code

property BMonthEnd.**rule_code**

Methods

BMonthEnd.apply(self, other)	
BMonthEnd.apply_index(self, other)	Vectorized apply of DateOffset to DatetimeIndex, raises NotImplentedError for offsets without a vectorized implementation.
BMonthEnd.copy(self)	
BMonthEnd.isAnchored(self)	
BMonthEnd.onOffset(self, dt)	
BMonthEnd.is_anchored(self)	
BMonthEnd.is_on_offset(self, dt)	
BMonthEnd.rollback(self, dt)	Roll provided date backward to next offset only if not on offset.
BMonthEnd.rollforward(self, dt)	Roll provided date forward to next offset only if not on offset.
BMonthEnd.__call__(self, other)	Call self as a function.

pandas.tseries.offsets.BMonthEnd.apply

BMonthEnd.**apply**(*self*, *other*)

pandas.tseries.offsets.BMonthEnd.apply_index

BMonthEnd.**apply_index**(*self*, *other*)

Vectorized apply of DateOffset to DatetimeIndex, raises NotImplentedError for offsets without a vectorized implementation.

Parameters

i [DatetimeIndex]

Returns

y [DatetimeIndex]

pandas.tseries.offsets.BMonthEnd.copy

BMonthEnd.**copy**(*self*)

pandas.tseries.offsets.BMonthEnd.isAnchored

BMonthEnd.**isAnchored**(*self*)

pandas.tseries.offsets.BMonthEnd.onOffset

BMonthEnd.**onOffset**(*self*, *dt*)

pandas.tseries.offsets.BMonthEnd.is_anchored

BMonthEnd.**is_anchored**(*self*)

pandas.tseries.offsets.BMonthEnd.is_on_offset

BMonthEnd.**is_on_offset**(*self*, *dt*)

pandas.tseries.offsets.BMonthEnd.rollback

BMonthEnd.**rollback**(*self*, *dt*)

Roll provided date backward to next offset only if not on offset.

Returns

TimeStamp Rolled timestamp if not on offset, otherwise unchanged timestamp.