

# Obscure holidays

**Square Root Day** is an unofficial holiday celebrated on dates where both the day of the month and the month are the square root of the last two digits of the year. For example, the last Square Root Day was celebrated on March 3, 2009 (3/3/09) and the next Square Root Day will be observed on April 4, 2016 (4/4/16). The final Square Root Day of the century will occur on September 9, 2081 (9/9/2081).

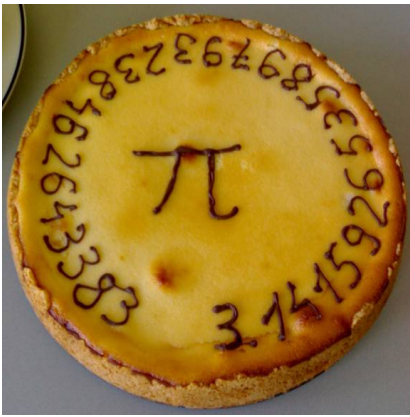
Ron Gordon – a Redwood City (California, USA) high school teacher – established the first Square Root Day on September 9, 1981 (9/9/81). Gordon remains the holiday's publicist, sending news releases to world media outlets. Gordon's daughter set up a [Facebook group](#) where people can share how they were celebrating the day. One suggested way of celebrating the holiday is by eating square radishes, or other root vegetables cut into shapes with square cross sections, thus creating a "square root".



The last Square Root Day was celebrated on March 3, 2009 (3/3/09).

But Square Root Day is definitely not the only obscure holiday. **Pi Day** ( $\pi$ -day) is an annual celebration commemorating the mathematical constant  $\pi$ . Pi Day is observed on March 14 (or 3/14 in the *month/day* date format), since 3, 1, and 4 are the first three significant digits of  $\pi$  in decimal form. Celebration usually starts at 1:59 p.m. since the six-digit approximation of  $\pi$  is 3.14159.

In the year 2015, Pi Day will have special significance on 3/14/15 at 9:26:53 a.m. and p.m., with the date and time representing the first 10 digits of  $\pi$ . That same second will also contain a precise instant corresponding to all of the digits of  $\pi$ . But the ultimate  $\pi$ -moment was celebrated on March 14, 1592 at 6:53:58, since 3/14/1592 6:53:58 corresponds to the first twelve digits of  $\pi$ . Pi Day has been observed in many ways, including eating pie, throwing pies, discussing the significance of the number  $\pi$  or organizing Einstein look-alike contests (Albert Einstein was born March 14, 1879).



*Pi pie* as it is served in many schools and universities on March 14.

And what do you think about the annual holiday celebrated on the last Friday in July? There's something unnerving about standing silently in a crowded elevator. It's a brave person who breaks the uncomfortable silence – so **Talk In An Elevator Day** is dedicated to plucking up our courage and making polite small talk whilst riding the elevator.

## Assignment

Your task is to determine the next holiday that follows a given date. In order to do so, you first have to implement the following three functions that each take a date (a `datetime.date` object) as their argument. Each function must return a Boolean value that indicates whether or not a particular (obscure) holiday is observed at the given date.

1. `piDay`: indicates whether the date is March 14
2. `squareRootDay`: indicates whether the day of the month and the month are the square root of the last two digits of the year
3. `elevatorDay`: indicates whether the date is the last Friday in July

Now, also implement a function `nextHoliday` that has the following parameters:

- a mandatory parameter `holiday` that takes a function object; in calling this function, a date (a `datetime.date` object) must be passed and a Boolean value is returned that indicates whether or not a particular holiday is observed on the given date
- an optional parameter `date` that takes a date (a `datetime.date` object)
- an optional parameter `self` that takes a Boolean value (default value `False`)

The function must return the date (a `datetime.date` object) the next holiday of the given type is observed that follows the given date. If no date is given, the next holiday should follow today's date. Holidays are determined as dates for which the given function returns the value `True`. In case the value `True` is passed to the parameter `self`, the given date should also be taken into account in determining the next holiday. Otherwise, the given date should not be taken into account in determining the next holiday.

## Preparation

In this assignment you will have to make use of the data types `date` and `timedelta` that are defined in the `datetime` module of the [Python Standard Library](#). Before starting to work on the actual assignment, you should first have a look at how Python responds when you execute the following sequence of instructions in an interactive Python session:

1. >>> from datetime import date  
>>> birthday = date(1983, 1, 14)  
>>> d = date.today() - birthday  
>>> type(d)  
>>> d.days
2. >>> from datetime import timedelta  
>>> birthday + timedelta(1)  
>>> day1 = birthday + timedelta(1)  
>>> day1  
>>> day2 = day1 + timedelta(1)  
>>> day2
3. >>> today = date.today()  
>>> today  
>>> today.weekday()  
>>> tomorrow = today + timedelta(1)  
>>> tomorrow.weekday()  
>>> tomorrow.day

Make sure that you understand why the different results are generated.

## Example

```
>>> import datetime

>>> piDay(datetime.date(2014, 3, 14))
True
>>> piDay(datetime.date(2021, 5, 24))
False

>>> squareRootDay(datetime.date(2016, 4, 4))
True
>>> squareRootDay(datetime.date(2081, 9, 9))
True
>>> squareRootDay(datetime.date(2014, 4, 5))
False

>>> elevatorDay(datetime.date(2014, 7, 25))
True
>>> elevatorDay(datetime.date(2015, 7, 21))
False

>>> nextHoliday(holiday=squareRootDay)
datetime.date(2025, 5, 5)
>>> nextHoliday(holiday=piDay)
datetime.date(2024, 3, 14)
>>> nextHoliday(elevatorDay, date=datetime.date(2014, 10, 12))
datetime.date(2015, 7, 31)
>>> nextHoliday(piDay, date=datetime.date(2015, 3, 14), self=True)
datetime.date(2015, 3, 14)
>>> nextHoliday(date=datetime.date(2081, 9, 9), holiday=squareRootDay)
datetime.date(2101, 1, 1)
```

**Vierkantwoordeldag** is een officieuze feestdag die gevierd wordt op datums waarop zowel de dag van de maand als de maand zelf de vierkantwoordel zijn van de laatste twee cijfers van het jaar. Zo viel de laatste vierkantwoordeldag bijvoorbeeld op 3 maart 2009 (3/3/2009) en wordt de volgende vierkantwoordeldag gevierd op 4 april 2016 (4/4/2016). De laatste vierkantwoordeldag

van deze eeuw valt op 9 september 2081 (9/9/2081).

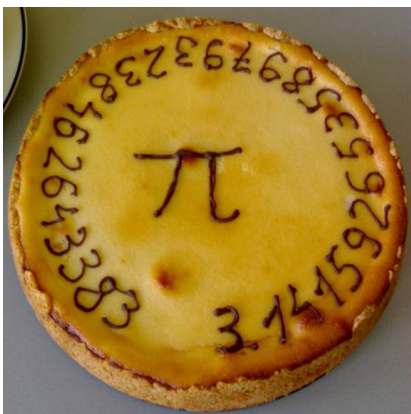
De eerste vierkantsworteldag werd op 9 september 1981 (9/9/1981) in het leven geroepen door Ron Gordon, een leraar uit Redwood City (Californië) in de Verenigde Staten van Amerika. Gordon blijft tot op vandaag deze feestdag promoten door persberichten te versturen naar allerlei media. De dochter van Gordon heeft een [Facebookgroep](#) opgericht waarop mensen kunnen delen hoe ze vierkantsworteldag gevierd hebben. Een aanbevolen manier om de feestdag te vieren is trouwens door het eten van vierkante radijzen of andere wortelgroenten die in vierkante stukken gesneden zijn en zo dus "vierkante wortels" vormen.



De laatste vierkantsworteldag werd gevierd op 9 maart 2009 (9/9/2009).

Maar vierkantsworteldag is zeker niet de enige obscure feestdag. **Pi-dag** ( $\pi$ -dag) is een feestdag die gewijd is aan de wiskundige constante  $\pi$ . Deze feestdag wordt wereldwijd op 14 maart gevierd, omdat deze datum in de Amerikaanse notatie geschreven wordt als 3/14 wat overeenkomt met de driecijferige benadering voor  $\pi$ . De viering begint gewoonlijk om 13:59 (1:59 PM) omdat 3,14159 de zescijferige benadering van  $\pi$  is.

Het ultieme  $\pi$ -moment viel te beleven op 14 maart 1592 om 6:53 en 58 seconden, wat overeenkomt met 3/14/1592 6:53:58 in de Amerikaanse notatie voor datums. Dit correspondeert met de eerste 12 cijfers van  $\pi$  (3,14159265358). Op scholen en universiteiten in Amerika begon men een aantal jaren geleden  $\pi$ -dag te vieren met het eten van taart (in het Engels *pie*; ook  $\pi$  wordt door Engelstaligen zo uitgesproken) of  $\pi$ -zza, worden er etenswaren gemaakt in de vorm van  $\pi$ , en organiseert men Einstein look-alike wedstrijden (Albert Einstein werd geboren op 14 maart 1879).



*Pi pie* ofwel  $\pi$ -taart zoals die typisch gebakken wordt op  $\pi$ -dag.

En wat dacht je van de feestdag die jaarlijks gevierd wordt op de laatste vrijdag van juli. Krijg je het steeds benauwd als je in een overvolle lift staat waarin niemand iets zegt? Vind je het ook prettig als iemand in een dergelijke situatie een poging doet om de stilte te doorbreken? Op de

**Talk In Elevator Day** wordt iedereen aangespoord de moed bijeen te rapen om een vriendelijk woord te richten aan zijn of haar medepassagiers tijdens het nemen van een lift.

## Opgave

Je opdracht bestaat erin de eerste feestdag te bepalen die volgt op een gegeven datum. Hiervoor moet je eerst de volgende drie functies schrijven waaraan telkens een datum (een `datetime.date` object) moet doorgegeven worden. Elke functie moet een Booleaanse waarde teruggeven die aangeeft of de gegeven datum al dan niet op een bepaalde (officieuze) feestdag valt.

1. `pidag`: geeft aan of de gegeven datum valt op 14 maart
2. `vierkantworteldag`: geeft aan of voor de gegeven datum de dag van de maand en de maand zelf de vierkantwortel zijn van de laatste twee cijfers van het jaar
3. `liftdag`: geeft aan of de gegeven datum valt op de laatste vrijdag van juli

Schrijf nu ook een functie `volgendeFeestdag` met de volgende parameters:

- een verplichte parameter `feestdag` waaraan een functie-object moet doorgegeven worden; bij het aanroepen van deze functie moet een datum (een `datetime.date` object) doorgegeven worden en wordt een Booleaanse waarde teruggegeven die aangeeft of de gegeven datum al dan niet op een bepaalde feestdag valt
- een optionele parameter `datum` waaraan een datum (een `datetime.date` object) kan doorgegeven worden
- een optionele parameter `zelf` waaraan een Booleaanse waarde kan doorgegeven worden (standaardwaarde `False`)

De functie moet de datum (een `datetime.date` object) teruggeven van de eerste feestdag die volgt op de gegeven datum. Indien er geen datum opgegeven wordt, dan moet de eerste feestdag teruggegeven worden die volgt op de datum van vandaag. Feestdagen worden bepaald als datums waarvoor de gegeven functie de waarde `True` teruggeeft. Indien aan de parameter `zelf` de waarde `True` wordt doorgegeven, dan moet de gegeven datum zelf ook in rekening gebracht worden als potentieel volgende feestdag. Anders wordt de gegeven datum buiten beschouwing gelaten bij het bepalen van de volgende feestdag.

## Vorbereiding

In deze opgave moet je gebruik maken van de gegevenstypes `date` en `timedelta` die gedefinieerd worden in de module `datetime` van de [Python Standard Library](#). Voor je aan de eigenlijke opgave begint, kan je best eerst nagaan hoe Python reageert als je achtereenvolgens de volgende instructies uitvoert binnen een interactieve Python sessie:

1. 

```
>>> from datetime import date
>>> geboortedatum = date(1983, 1, 14)
>>> d = date.today() - geboortedatum
>>> type(d)
>>> d.days
```
2. 

```
>>> from datetime import timedelta
>>> geboortedatum + timedelta(1)
>>> dag1 = geboortedatum + timedelta(1)
>>> dag1
```

```
>>> dag2 = dag1 + timedelta(1)
>>> dag2
```

```
3. >>> vandaag = date.today()
>>> vandaag
>>> vandaag.weekday()
>>> morgen = vandaag + timedelta(1)
>>> morgen.weekday()
>>> morgen.day
```

Zorg er zeker voor dat je begrijpt waarom de verschillende resultaten gegenereerd worden.

## Voorbeeld

```
>>> import datetime
```

```
>>> pidag(datetime.date(2014, 3, 14))
True
>>> pidag(datetime.date(2021, 5, 24))
False
```

```
>>> vierkantwoordtdag(datetime.date(2016, 4, 4))
True
>>> vierkantwoordtdag(datetime.date(2081, 9, 9))
True
>>> vierkantwoordtdag(datetime.date(2014, 4, 5))
False
```

```
>>> liftdag(datetime.date(2014, 7, 25))
True
>>> liftdag(datetime.date(2015, 7, 21))
False
```

```
>>> volgendeFeestdag(feestdag=vierkantwoordtdag)
datetime.date(2025, 5, 5)
>>> volgendeFeestdag(feestdag=pidag)
datetime.date(2024, 3, 14)
>>> volgendeFeestdag(liftdag, datum=datetime.date(2014, 10, 12))
datetime.date(2015, 7, 31)
>>> volgendeFeestdag(pidag, datum=datetime.date(2015, 3, 14), zelf=True)
datetime.date(2015, 3, 14)
>>> volgendeFeestdag(datum=datetime.date(2081, 9, 9), feestdag=vierkantwoordtdag)
datetime.date(2101, 1, 1)
```