

ScaligerFX

Calendrical Calculations



2016 Albert Garre



Joseph Justus Scaliger (August 5, 1540 - January 21, 1609) introduced the Julian Day Count, starting with January 1, 4713 BCE 12:00 as Julian Day 0, which is widely used for astronomical calculations and is also the basis for the calendrical calculations performed by this program. So it seems justified to dedicate this program to J. J. Scaliger, who lived in France and the Netherlands. You may find some details about his life in Wikipedia: http://en.wikipedia.org/wiki/Joseph_Justus_Scaliger.

The portrait of Joseph Justus Scaliger is reproduced here with kind permission of the *Leiden University Library* (Special Collections - Collection Services Department). You may find it on the homepage of the Scaliger Institute of the Leiden University <http://www.bibliotheek.leidenuniv.nl/bijzondere-collecties/scaliger-instituut/over-scaliger-instituut/over.html>.

History:

ScaligerFX is an adaptation of the former program *Scaliger* to the JavaFX 8 environment.

Installation and start:

ScaligerFX requires the **Java SE Runtime Environment 8u40** (or newer version).

Just extract the files from the downloaded archive into any directory. If you want to make the program useable for all users, take a subdirectory of `\users\public` (Windows) resp. of `/opt` (Linux) and set the access rights (read and execute) accordingly. Files needing write permission by the program are copied at first start to a hidden directory `.scaligerfx` in `\users\[user]` (Windows) resp. `/home/[user]` (Linux). All configuration files are plain UTF-8 text files and may be edited.

In Windows the program is started by **DOUBLE-Click** on the file **ScaligerFX.jar**. In Unix it may be necessary to configure a program starter to allow this (see for example <https://www.youtube.com/watch?v=B6S7tKEZ1a4>).

If you want to use another language than the default language of your system, you may start the program from the cmd-line with e.g. **java -jar ScaligerFX.jar --lan=De** (for German language)

Calendar

Internally the program uses the Julian Day Count (see above). The Julian Day is converted into a date following the rules of a mixed Julian/Gregorian Calendar: Dates before 4.10.1582 use a (proleptic) Julian Calendar, dates after 15.10.1582 use the Gregorian Calendar. This matches with Wikipedia, when displaying person data.

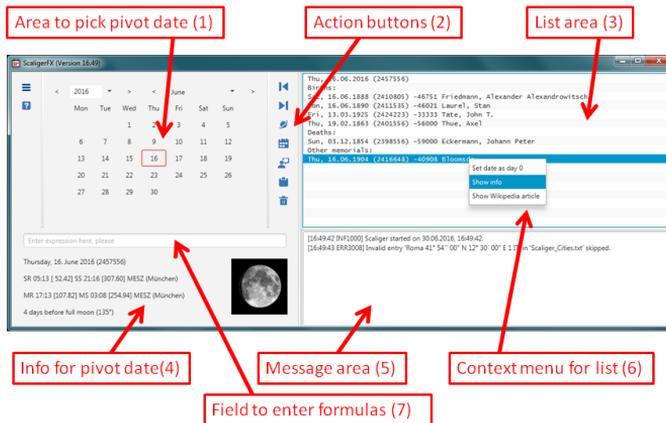
In contrast to the "genuine" Julian Day Count, used by astronomers, the Julian Day in the program starts at 0:00 and ends at 23:59. The genuine JD starts at 12:00 and ends at 11:59 the next day.

The proleptic Julian Calendar used in the Julian Day Count provides a continuous backward year count over year 0 to year -1 and so on. The Julian calendar used by historians knows of no year 0 and jumps from year 1 CE backward to year 1 BCE. So 'proleptic year' -1 is identical with 'year of era' 2 BCE and Caesar was murdered in the 'proleptic year' -43 (44 BCE). As a consequence a year is a leap year, if the 'proleptic year' can be divided by 4 (for dates before 1582), not if the 'year in era' is divisible by 4. So year -8 (year 9 BCE) is a leap year but year 8 BCE (-7 proleptic) is not.

The dates displayed by the program are generally formatted as 'year of era' and suffixed by 'BCE' if necessary. The only exception is the field to enter the year in **(1)** (see below), where a proleptic year is requested and also displayed - otherwise the function of the **[<]**- resp. **[>]**-Buttons for the year would be a bit confusing. For the input of years as part of a formula in **(7)** (see below), both possibilities are allowed: either -yyyy (proleptic) or yyyyB as year BCE). See also the chapter *Advanced Computations* below.

Note that for years greater or equal to 1 there is no difference between 'proleptic year' and 'year in era'.

Overview:



Basic Functions:

Select a date using the Calendar in the area (1). You may enter a year (proleptic year) between -4712 and 9999 into the corresponding field, select a month from the second box and a day by clicking the *Buttons* with the day numbers. The buttons [**<**] resp. [**>**] act on the field resp. box next to them and count down and up the year resp. month by one. bzw. hoch.

The info in (4) is updated immediately and gives you the Julian Day, the time for sunrise, sunset, moonrise and moonset and a picture of the current phase of the moon. To adjust the astronomical data for your location you have to select the appropriate location in the options dialog (see below). You may add more locations to the file *Scaliger_cities.txt*, using any UTF8-text-editor (e.g. Notepad). The date displayed on the left side is called the *pivot date*.

Buttons:



Click here to transfer the *pivot date* to the list area (3) on the right side.



Click here to make the date selected in the list area (3) the new *pivot date*.



Click here to transfer the *pivot date* to the list area (3) including astronomical data. Pressing the **SHIFT** key while clicking will generate a list with astronomical data for the *pivot month*. Output is one line for every date. The list area is cleared before the new list is build.



Click here to generate a list, by adding 1000 days, 1111 days , 2000 days etc. to the pivot date. Output is one line for every date in the list area (3). The list area is cleared before the new list is build. With **SHIFT-Click** on this button the ages of people found in the files *Scaliger_Names.txt* (see next chapter) are also added to the pivot date. Try this with your birthday and you will get a list, telling you, when you will surpass the age of e.g. Isaac Newton.



Click here to generate a list with memorials for the *pivot date*, e.g. entries of people who were born or have died 10000 days, 20000 days etc. before the pivot date. The dates are read from the files *Scaliger_Names_xxx.txt* in the installation directory (see next chapter). The list area (3) is cleared before the new list is build. A short way to generate this list, is to **Click** on an already selected day number (marked by a red border). If you use **SHIFT-Click** the Button, the memorials for the whole month are listed in the list area (3).



Click here to copy the content of the list area (3) to the Clipboard.



Click here to clear the list area (3).



Click here to open a window (*option window*) which allows to change some settings. The handling of the option window should be

self-explanatory and will not be further described. Changed settings are saved in the file *config.xml* and evaluated at next program start.

The list area (3):

You may select every line of the list area (3) by clicking on it. If the selected line contains a date, it can be made the new pivot date by clicking on the *Left Arrow Button* (see above). A **RIGHT-Click** into the list area opens a context menu (6), which offers additional actions for the selected date:

Set as day 0: The selected date is taken as day 0 and for every other date within the list area the time span (in days) between this date and the selected date will be shown.

Show info: Additional information about the selected date is shown in a small dialog window (if the date is connected with a person or an event).

Show Wikipedia article: Will start the default browser to display the Wikipedia-entry for a person or event connected with the selected date (if available).

The files *Scaliger_Names_xxx.txt*

These files must be located in the same directory as the *jar-File* of the program. They are extracted from Wikipedia and contain information about important persons and events (name, date of birth, date of death, date of event, short information about profession etc.). The persons are sorted into several files for persons born in different centuries.

The files can be edited with any UTF8-text-editor (e.g. Notepad). A detailed description, how to generate and/or edit these files is given in *Scaliger_Names.txt*, which is the central administration file for all the Files *Scaliger_Names_xxx.txt*.

Advanced Computations with dates:

The field (7) below the date picker may be used to enter expressions for advanced computations.

(Almost) every arithmetic expression is allowed. Evaluation starts after **ENTER** is hit. The further processing of the result depends on the *prefix* of the expression:

- ~ **No prefix:** The expression is evaluated and the result is displayed in the message area (5).
- ~ **Prefix '!':** The expression is evaluated and the result is

displayed in the message area (5) together with the prime factorization of the result (for results < 100000001).

- ~ **Prefix '=':** The result is taken as a Julian Day and the corresponding date is made the new *pivot date*.
- ~ **Prefix '+':** The result is taken as a number of days and added to the current pivot date.
- ~ **Prefix '-':** The result is taken as a number of days and subtracted from the current pivot date.
- ~ **Prefix 'J':** The result is taken as a Julian Day and displayed in the list area (3) without changing the pivot date..
- ~ **Prefix '[':** A loop construct of the form [x=a,b] is expected. Instead of ,x' other variables may be used. The specified variable will be replaced successively by a, a+1, a+2 till b in the expression following the loop construct. The results will be taken as Julian Day and displayed in the list area (3).

Expressions with prefix] and *loop constructs* may contain a comment (included in apostrophs). This comment will also be displayed in the list area (3) (loop variables within the comment are replaced by their proper value) and has the additional effect, that no attempt is made to interpret the result(s) as Julian Day.

- ~ **Prefix '~':** The result is taken as a number of days and a list of people is displayed who lived "about" as long as specified by this number (the meaning of "about" may be configured via the *option window*). Whether the list area (3) is cleared before displaying the new list may also be specified via the option window.
- ~ **Prefix '%':** The result is taken as a Julian Day and a list of memorials (births, deaths, other events) is displayed, which took place on a day with "about" the same phase of the moon as the result of the expression (the meaning of "about" may be configured via the *option window*). The list area (3) is cleared before displaying the new list.
- ~ **Prefix '>':** The result is taken as a number of days and a list of people is displayed who lived longer as specified by this number. Whether the list area (3) is cleared before displaying the new list may be specified via the option window.
- ~ **Prefix '<':** The result is taken as a number of days and a list of people is displayed who lived shorter as specified by this number. Whether the list area (3) is cleared before displaying the new list may be specified via the option window.
- ~ **Prefix '?':** If ? is followed by a string, the string is taken as a part of a name of a person or of some event. The files *Scaliger_Names_xxx.txt* are searched and the dates of

matching entries are displayed in the list area (3). If ? is followed by § or ° and a date like *mm.yyyy* the files *Scaliger_Names_xxx.txt* are searched for persons who died or are born in the month *mm.yyyy*. If no month is specified all persons who died or are born in the year *yyyy* are listed.

- ~ **Prefix '\$'**: (Only valid with a string). The string is taken as a part of a name of a person or of some event. The list area (3) is searched for an entry containing this string and the first matching line is selected. The search starts after the currently selected line, so that multiple matches can be displayed by repeatedly pressing **ENTER**.

Within an expression numbers may be given explicitly or symbolically. The following symbols are allowed:

- ~ **!!** will be replaced by the the Julian Day Number for the pivot date
- ~ **dd.mm.yyyy**: will be replaced by the Julian Day Number for this date (if it's a valid date). Instead of a number for **dd** one of the symbols **su1**, **su2**, ..., **sux**, **mo1**, **mo2**, ... , **mox** , ... , **sax** for the 1st, 2nd, last Sunday, ... , Saturday of the month may be specified . The abbreviations of the day of week may be specified in upper- or lower case and regardless of the choice of language also in German (see German helptext). Years BCE may be specified either 'proleptic' with negative sign (**-yyyy**) or as 'year of era' with suffix B (**yyyyB**). Note that the year **-x** corresponds to the year **(x-1)B** and vice versa.
- ~ **_GY**: will be replaced by the length of the Gregorian year 365.2425 d.
- ~ **_JY**: will be replaced by the length of the Julian year 365.25 d.
- ~ **_SM**: will be replaced by the length of the synodic month (lunar cycle) 29.530589 d.
- ~ **Easter.yyyy**: will be replaced by the Julian Day Number of Easter in the specified year. Abbreviation (min 3 letters) and mix of upper- / lower case letters are allowed. Regardless of the choice of language, the German name of the feast (see German helptext) is allowed too.
- ~ **Ascension.yyyy** will be replaced by the Julian Day Number of the Ascension Day in the specified year. Abbreviation (min 3 letters) and mix of upper- / lower case letters are allowed. Regardless of the choice of language, the German name of the feast (see German helptext) is allowed too.
- ~ **Pentecost.yyyy** will be replaced by the Julian Day Number of Pentecost in the specified year. Abbreviation (min 3 letters) and mix of upper-/lower case letters are allowed. Regardless of the choice of language, the German name of the feast (see German helptext) is allowed too.
- ~ **°name**: will be replaced by the Julian Day Number of the

birthday of the specified person. The name must identify the person uniquely within files *Scaliger_Names_xxx.txt*. Underscores within the name will be replaced by blank.

- ~ **\$name**: will be replaced by the Julian Day Number of the day when the specified person died. The name must identify the person uniquely within files *Scaliger_Names_xxx.txt*. Underscores within the name will be replaced by blank. If the person is still living, evaluation of the expression is cancelled with syntax error.
- ~ **&name**: will be replaced by the age of the specified person in days. If the person is still living, the age is computed relative to *!Today* (see below-). The name must identify the person uniquely within files *Scaliger_Names_xxx.txt*. Underscores within the name will be replaced by blank.
- ~ **!name**: will be replaced by the Julian Day Number of a "event" listed in *Scaliger_Names_Static.txt*. The special event *!Today* is used for the current date. Underscores within the name will be replaced by blank.

Examples:

It is assumed that all names used in the examples are specified as "relevant" by an entry in *Scaliger_Names.txt* (see comment in that file)..

=11.11.1111 sets the pivot date to this date.

=2127165 sets the pivot date to this Julian Day (which happens to be the 11.11.1111).

?Gau searches for entries in *Scaliger_Names_xxx.txt* containing "Gau", will display the birthday of C.F.Gauß (among others).

?\$1855 will list all Persons, who died in 1855 (among them Gauß).

?°4.1707 will list all Persons, born in April 1707 (among them Euler).

°Gauß - °Euler computes how many days after Eulers birthday Gauß was born (25583).

(°Gauß - \$Euler) / 365 computes how many years after Eulers death Gauß was born (-6). If you want higher precision, you may of course use **_GY** as divisor.

°Gauß-asc.1777 computes the number of days between Ascension in 1777 and Gauß' Birthday (which is -8 and so corresponds with the statement of Gauß' mother, that Carl Friedrich was born "on wednesday in the week ascension").

~&Gauß displays a list of people who became about as old as Gauß (e.g. Galilei who died after 28442 days, Gauß passed away after

28422 days).

+(!Today - °Gauß) adds the time span since Gauß' birth to today's date (no idea if this makes any sense).

=°Gauß+&Galois displays the date when Gauß surpassed the age of Galois (5. Dec. 1797).

%°Gauß displays all events that took place on a day with about the same phase of the moon (275° or 7 days before new moon) as Gauß' birthday (e.g. Hilbert's birthday).

/&Schafarewitsch displays the age of the deceased russian Mathematician Schafarewitsch in days and its prime factorization. To the amazement and delight of every mathematician the result is $2 \times 3 \times 5 \times 7 \times 163$.

=°Bernoulli is not unique and will result in an error message – there are entries for Jakob I., Jakob II. and Johann Bernoulli. To find the last one, you have to specify something like **=°Bernoulli,_Jo** (with comma and underscore).

[x=2013,2040]easter.x displays all easter dates for the years 2013 to 2040 in the list area **(3)**.

[x=1,12][y=2013,2015]13.x.y displays for all months of the years 2013 to 2015 the 13. day.

[x=2000,2020]su2.5.x displays for each of the years 2000 to 2020 the date of the 2nd Sunday in May (Mother's Day).

If you want to compute the day, when the sum of the ages of the members of your family (Ann, Bert and Claire) add up to 100 years, enter **=(_GY * 100 + °Ann + °Bert + °Claire) / 3** (check the formula!). This will work, if you added the birthdays of Ann, Bert and Claire to *Scaliger_Names_Static.txt* and the names Ann, Bert and Claire are unique within the files *Scaliger_Names_xxx.txt*. If Ann and Bert are **much** older than Claire, the date may lie **before** Claire's birthday ;-)

License:



This program is free software due to . Details can be found at <https://creativecommons.org/licenses/by-nc-sa/3.0/>. The program is written with Java SE using Netbeans..

Source of the used pictures:

The Pictures of the moon phases are from http://tycho.usno.navy.mil/gif/moons/m***.gif where *** is the three digit number of the phase in degrees, starting with 000 (new moon).

Used packages:

The program uses the package *org.threeten.extra.chrono*, which requests the publication of the following notice:

```
/*
 * Copyright (c) 2007-present, Stephen Colebourne & Michael Nascimento Santos
 *
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions are met:
 *
 * * Redistributions of source code must retain the above copyright notice,
 *   this list of conditions and the following disclaimer.
 *
 * * Redistributions in binary form must reproduce the above copyright notice,
 *   this list of conditions and the following disclaimer in the documentation
 *   and/or other materials provided with the distribution.
 *
 * * Neither the name of JSR-310 nor the names of its contributors
 *   may be used to endorse or promote products derived from this software
 *   without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
 * "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
 * LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
 * A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR
 * CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
 * EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
 * PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
 * PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
 * LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
 * NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
 * SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 */
```