



I'm not robot



Open

platform's idea of local time. The return values of gmtime(), localtime(), and strptime() also offer alternative names for individual fields. CLOCK_UTCIME_RAWÂA Clock that increments monotonically, tracking the time since an arbitrary point, unaffected by time adjustments and not incremented when the system is asleep. time.CLOCK_PROFÂA High-resolution per-process timer from the CPU. if 2006 < year < 2007: dststart, dstend = DSTSTART 2007, DSTEND 2007 elif 1966 < year < 2007: dststart, dstend = DSTSTART 1987, 2006, DSTEND 1987, 2006 elif 1966 < year < 1987: dststart, dstend = DSTSTART 1967, 1986, DSTEND 1967, 1986 else: return (datetime(year, 1, 1), 1) * 2 start = first_sunday_on or after(dststart.replace(year=year)) end = first_sunday_on or after(dstend.replace(year=year)) return start, end class USTimeZone(zinfo): def __init__(self, hours, reprname, stdname, dstname): self.stdoffset = timedelta(hours=hours) self.reprname = reprname self.stdname = stdname self.dstname = dstname def __repr__(self): return self.reprname def tzname(self, dt): if self.dst(dt): return self.dstname else: return self.stdname def utcoffset(self, dt): return self.stdoffset + self.dst(dt) def dst(self, dt): if dt is None or dt.tzinfo is None: # An exception may be sensible here, in one or both cases. time.thread_time() ¶ÂA float¶A Return the value (in fractional seconds) of the sum of the system and user CPU of the current thread. time.process_time_ns() ¶A int¶A rotation Similar to process_time() but return time as nanoseconds. The format parameter uses the same guidelines used by strftime(); the default is "%a %b %d %H:%M:%S %Y" which matches format returned by ctime(). monotonic: True if the clock cannot go backward. False otherwise resolution: The resolution of the clock in seconds (float) time.gmtime(secs)¶A Converts a time expressed in seconds since the start of the start to a struct_time in UTC where the dst flag is always zero. The argument can be a floating point number to indicate a more accurate sleep time. It returns a floating-point number, for compatibility with time(). time() The inverse of gmtime() of this ¶. Here is an example, a format for dates that is compatible with the one specified in the RFC 2822 Internet E-mail Standard. If the string cannot be parsed according to the format, or if you have excessive data after ¶, the ValueError is elevated. %W Number of the week of the year (Monday as first day of the week) as a decimal number [00.53]. (2) %U Number of the week of the year (Sunday as first day of the week) as a decimal number [00.53]. On the other hand, the precision of time() and sleep() are better than their Unix equivalents: times are expressed as floating-point numbers, time() returns the most accurate time available (using gettimeofday()) Unix when available), and sleep() will accept a time with a non-zero fraction (Unix select) used to implement this when available). When 2-digit years are analyzed, they are converted according to POSIX and ISO C standards: values 69-99 are mapped to 1969-1999, and values 00-68 are mapped to 2000-2068. They are commonly referred to as Unix time. %d Mother's Day as a decimal number [01.31]. Use threading.get_ident() or the ident attribute of the thread thread objects to get a suitable value for res_res_edop_laer_opmet_sep_ÂAñuf_sair;_Áv_sad_of_ÂAícerp_ÂA than suggested by the units in which its value or argument is expressed. Also refer to the object-oriented interface of the datatimemore module for dates and hours. Sometimes it can be useful to consult the documentation of the platform because the semi-sq of these functions varies between the platforms. >>> OS.Environ ['tz'] = 'EST + 05EDT, M4.1.0, M10.5.0' >>> Time.tzset () >>> Time.StrTime ("% x% x% z") '02:07:36 05/08/03 edt' >>> OS.Environ ['tz'] = 'AEST-10AEDT-11, M10.5.0, M3.5.0, M3.5.0' >>> Time.tzset () >>> time.sttime ("% x% x% z") '16: 08: 12 05/08/08 AEST' on many UNIX systems (including * BSD, Linux, Solaris and Darwin), is more convenient Use the Zlonifo database (Tzfile (5)) to specify the time zone rules. Time.monotonic () ¶ ¶ "Returns the value (in fractional seconds) of a monotonic clock, this is, a clock that can not go back. See above for a description of the Structur_time_object. Use process_time_ns () to avoid loss of accuracy caused by float type. A value of -1 indicates that this is not known and usually results in the correct state being filled. Availability: UNIX (see man page for PTREAD_GETCPUCKLOCKID (3) for more information). For related functionality, refer to the datetime and calendar modules as well. Example of symbol% a Sunday * W 0..6 (Sunday is 0) Y 13 Year% y 2013 * B Jan MÃs% B January * M 01.31 Day % And 1..31 , day example of symbol to rea% l 1 hour% h 00..23 24h hour% i 01..12 12h hour% m 00..60% s 00..60 second p is or pm% z +08 time spindle% j 001..366 year of the year %% literal% character time.js format examples example of sauce yyyy-mm-dd 2014-01-01 dddd, mmmm fair YYYY, MAY 16, 2014 DDDD [o] MMMM MMMM Friday, May 16th Example of HH Saúda: MM at 12:30 pm used for moment.js and Data-FNS / format. For a complete and updated set of DST # and of time spindle, visit the Olson database (or try pytz): # # secivres noitazilanoinatretileacol eludom noituloser dnocesanon et of esolec ewig yam dna, ecruos erawdrah lamitno na esu ot stpmetta taht remit SERHGIIH KCOLC a sai SO siraloS eht Á e ASERHGIIH KCOLC.emit 00:00:20 si .nevig ton si emit fi .luafed eht .luaf noitatinatnemges sa heus .roivahab denifednu ni thuser yam di daerht derixpe ro dilavni na gnissP gninraW .sdnocesonan sa emit nruter tub .)(cintonom ot ralmiS Átni Á Á eht si hcihw .robotC fo yadnuS # tsal eht no)emit TSD(ma2 ta dne ot dna lirpA ni yadnuS # tsrif eht no)emit dradnats(ma2 ta tracs ot desu TSD. 6002 ot 7891 morF # j2 ,1 ,1 ,1 .1.(emitetad = 7002, DNETSd .rewen dna 21.01 SÓcam ýtilibaliavA rectal ro 93.6.2 xuniL ýtilibaliavA .)(emitteg kcolc dna)(serteg kcolc rof sretemarap sa desu era stnatsnoc esehT 'TSEE' , 'TEE'(emanzt.emit >>)(teszt.emit >>> 'tpygE' =]ZT[noirvne.so >>]TDE', 'TSE'(emanzt.emit >>)(teszt.emit >>> 'nretsae/SU' =]ZT[noirvne.so >> .elur dlof eht write :ruoh tnetsix-non a (paG # :RUOH + trats < td =< trats fi RUOH esle dlof.td fi OREZ nruter .fles si ofnizt.td htiw emitted a sessap)noitatnemelpmi #)(enozemitsa luafed eht yb dellac(noitatnemelpmi)(ctumorf # luafed 16,00[rebmun lamiced a sa dnoces S% j1(.no deiler eb ton dlouhs roivahab siht .)(teszt gnillac tuohtiw)(emitlacol ekil snoitcnuf fo tuputo eht tceffa yam elbairav tmemorivne ZT eht gnignahc .sesac ynam ni hguohtÁ etoN .ofnienoz/erahs/rsu/ ta detacol yllausu .esabatað zemit TSD ,7002 ecnis .SU eht nI # letad-ot-pu eb ton thgim(/ztyp/stcejorp/ten.egrofecruos//pPTH # emit = tt)(emitkm.emit = pmats j0 .0 .) (yadkeew.td ,dnoces.td ,etunim.td ,ruoh.td ,yad.td ,htnom.td ,raey.td(= tt):td ,fles(tsdsi fed)jtd(tsdsi .fles(emanzt.emit nruter :jtd ,fles(emanzt fed OREZ nruter :esle FFDITSD nruter :jtd(tsdsi .fles fi):td ,fles(tsd fed TESFFODTS nruter :esle TESFFOTSD nruter :jtd(tsdsi .fles fi):td ,fles(tesffoctu fed)dlof=dlolf ,fles=ofnizt .dnocesorim.td=dnocesorim .sgra*(emitetad nruter)jfid tsd - pmats(emitlacol.emit = sgra(= dlof dlof tceetD # DNOCES // FFDITSD = ffid tsd j0:)pmats(emitlacol.emit = sgra DNOCES //))fles=ofnizt ,1 ,1 ,0791(emitetad - tdt = pmats fles si ofnizt.td tressa :jtd ,fles(ctumorf fed .)ofnizt(enozem)TiAcOL ssaic TESFFODTS - TESFFOTSD = FFDITSD TESFFODTS = TESFFOTSD :esle jenoztia.emit - = sdnoces(atledemit = TESFFOTSD :thgilyad.emit fi jenozmit.emit - = sdnoces(atledemit = TESFFODTS emit .sa emit trompi)tsap eht ni degnahc # dah setur TSD eht ro/dna tesfo CTU erehw senozem# ni semit lacirotsih no seutlav gnorw ni thuser yam(# .atseo ÁA ,oirjArtnoc osac .lapicniP onaidireM od etsel ÁA ,oirjAroh osuf O ,eÁ-eÁ mu rop odidcerp eS .sdnugessonan moc aroh a enifed sam)(emittes kcolc a ralmiS ÁÁA)tni .emit .di klc(sn .emittes kcolc.emit .adij;Áv ÁA sadamahc saud ed sodatluser so ertne aSÁneretid a sanepa euq odom ed .odinifedni ÁA odanrotor rolav od aicnÁArefer ed otnop O .xinU .edadllibinopsiD .)0(emitng ajev .amrofatalp adad amun acup@ÁA ÁA laur rirbocsed araP .)(emitlacol uo)(emitng rop odanrotor omoc emit .tcurts nu ÁA onrotor ed rolav O .y% otamrof ed ogidÁÁe o jÁd es odnaug sotigÁd 2 ed sona rasilana edop)(emitprts ofÁÁAnuf A .rodautlif ed opti olep adasuae ofÁsicerp ed adrep a rative arap (sn .cintonom esU .sol-ÁtArt sereuq omoc ed ednepeD # .adasu res eved ofÁn aiedac adnuges a .odinifed revitse TSD ,oirjAroh osuf muhnen eS .ecapsemad ed otejbo mu omoc odacifcepsa oigÁÁer o erbos sepÁÁAmrofni ahnetB0 US time zones. start[time], end[time]Indicates when to change to and back from DST. PTP or NTP software can maintain a leap second table. Both string and format must be strings. CLOCK_HIGHRES is the nonadjustable, high-resolution clock. All days in a new year preceding the first Sunday are considered to be in week 0. They are shown without the optional field width and precision specification, and are replaced by the indicated characters in the strftime() result: Directive Meaning Notes %a LocaleÁÁAs abbreviated weekday name. The following constant is the only parameter that can be sent to clock_settime(). It is process-wide by definition. time.time() ¶A float¶A Return the time in seconds since the epoch as a floating point number. Leap days are counted, and it is possible to refer to February 29. return HOUR if dt.fold else ZERO # DST is off. It does include time elapsed during sleep and is system-wide. Changed in version 3.10: On macOS, the function is now system-wide. Availability: Windows, Linux, Unix systems supporting CLOCK_THREAD_CPUTIME_ID. For years before 1967, return # start = end for no DST. The earliest date for which it can generate a time is platform-dependent. %m Month as a decimal number [01.12]. For Unix, the epoch is January 1, 1970, 00:00:00 (UTC), time.sleep(secs)¶A Suspend execution of the calling thread for the given number of seconds. In calls to mktime(), tm_isdst may be set to 1 when daylight savings time is in effect, and 0 when it is not. time.altzone¶A The offset of the local DST timezone, in seconds west of UTC, if one is defined. %H Hour (24-hour clock) as a decimal number [00.23]. This is negative if the local DST timezone is east of UTC (as in Western Europe, including the UK). return HOUR if end - HOUR