



ΠΕΡΙΕΧΟΜΕΝΑ:

1. Η κλάση date
2. Η κλάση time
3. Η κλάση timedelta
4. Η κλάση datetime
5. Μέθοδοι Μορφοποίησης
Ημερομηνίας/Ωρας

Ευάγγελος Μ.

Σμαραγδένιος Χορηγός Μαθήματος

Πάνος Γ.

Ασημένιος Χορηγός Μαθήματος

Η κλάση date:

- Ορίζει μία ημερομηνία. **Κατασκευαστής:**

date(year, month, day)	Αρχικοποιεί το αντικείμενο
------------------------	----------------------------

- Εναλλακτικά η αρχικοποίηση μπορεί να γίνει και με τις:

Μέθοδος	Επεξήγηση
fromisoformat(string)	string==‘yyyy-mm-dd’
today()	Σημερινή ημερομηνία

- Το αντικείμενο που κατασκευάζεται είναι immutable και έχει μέλη:

- year: έτος
- month: μήνας
- day: ημέρα

Μέθοδοι:

- Υπάρχει μεγάλη ποικιλία μεθόδων. Κάποιες από αυτές είναι:

Μέθοδος	Επεξήγηση
isoformat()	στη μορφή: ‘yyyy-mm-dd’
replace(year=self.year, month=self.month, day=self.day)	Επιστρέφει νέο αντικείμενο date, αντικαθιστώντας τα ορίσματα με λέξεις κλειδιά με τις αντίστοιχες τιμές
weekday()	Ακέραιος: 0=Δευτέρα, 1=Τρίτη,...

- Επίσης:
 - Έχουν υλοποιηθεί μέθοδοι για πλήρη έλεγχο της εκτύπωσης (βλ. επόμενη διαφάνεια)
 - και έχουν οριστεί μαγικές μέθοδοι για **σύγκριση ημερομηνιών, εκτύπωση ως συμβολοσειρά**

Παράδειγμα 1: date/construct date.py

```
from datetime import date

date1 = date(2021, 11, 21)
print(date1)

date2 = date.today()
print(date2)
print(f'{date2.day}/{date2.month}/{date2.year}')

date3 = date.fromisoformat('2021-11-21')
print(date3)
```

Παράδειγμα 2: date/methods.py

```
from datetime import date

d = date(1990, 11, 1)
print(d.isoformat(), d)

d2 = d.replace(day=12)
print(d2)
print(f'Weekday: {d2.weekday()}')

print(d < d2)
print(d > d2)
```

ΜΑΘΗΜΑ 3.1: To module datetime

2. Η κλάση time

Η κλάση time:

- Ορίζει μία χρονική στιγμή ημέρας. **Κατασκευαστής:**

<code>time(hour=0, minute=0, second=0, microseconds=0)</code>	Αρχικοποιεί το αντικείμενο.
---	-----------------------------

- δυνατές τιμές 0:0:0.0 έως 23:59:59:999999
- Στο `immutable` αντικείμενο αποθηκεύονται τα μέλη, **`hour`**, **`minute`**, **`microseconds`**

Προχωρημένο:

- Στον παραπάνω κατασκευαστή υπάρχουν προαιρετικά ορίσματα που αφορούν την ζώνη ώρας (π.χ. ζώνη Ελλάδας GMT+2)
- (Θα το μελετήσουμε σε επόμενη διαφάνεια)

Άλλοι κατασκευαστές:

Μέθοδος	Επεξήγηση
<code>fromisoformat(string)</code>	<code>string == 'HH[:MM[:SS[.ffffff]]]'</code>

Παράδειγμα 3: time/construct time.py

```
from datetime import time

time1 = time(11, 14, 24)
print(time1)

time2 = time.fromisoformat('11:14:12')
print(time2.hour, time2.minute, time2.second)
```

Μέθοδοι:

Μέθοδος	Επεξήγηση
<code>isoformat()</code>	Επιστρέφει τη μορφή ISO της ώρας
<code>replace(hour=self.hour, minute=self.minute, second=self.second, microsecond=self.microsecond)</code>	Επιστρέφει νέο αντικείμενο time, αντικαθιστώντας τα ορίσματα με λέξεις κλειδιά με τις αντίστοιχες τιμές

- Επίσης έχουν οριστεί μαγικές μέθοδοι για σύγκριση χρονικών στιγμών και εκτύπωση ως συμβολοσειρά

Παράδειγμα 4: timedelta/methods.py

```
from datetime import time, timedelta

t1 = time.fromisoformat('11:12:13')
t2 = t1.replace(hour=23, second=11)
print(t1, t2)
print(t1 < t2)
print(t1 > t2)
```

Η κλάση timedelta:

- Ορίζει μία χρονική διάρκεια. **Κατασκευαστής:**

<code>timedelta(weeks=0, days=0, hours=0, minutes=0, seconds=0, microseconds=0, milliseconds=0)</code>	Αρχικοποιεί το αντικείμενο.
--	-----------------------------

- Στο `immutable` αντικείμενο αποθηκεύονται μόνο (αφού γίνουν κατάλληλοι υπολογισμοί) τα ακόλουθα (που είναι και μέλη της κλάσης):
 - `days`:** μέρες (εύρος $-10^9 \dots 10^9$)
 - `seconds`:** δευτερόλεπτα
 - `microseconds`:** 10^{-6} δευτερόλεπτα
- Έχουν οριστεί οι συνήθεις αριθμητικοί τελεστές (+, -, *, (με ακέραιο ή float), / (με ακέραιο ή float), //, %, -(προθεματικός))

Παράδειγμα 5: timedelta/diff.py

```
from datetime import timedelta

t1 = timedelta(hours=5)
t2 = timedelta(days=10, hours=25, seconds=100)
print(t1+t2)

print(t1/5, t1/100000)

print(-t2)
```

Μέθοδοι:

Μέθοδος	Επεξήγηση
<code>total_seconds()</code>	Συνολικό πλήθος δευτερολέπτων

- Επίσης έχουν οριστεί **μαγικές μέθοδοι για σύγκριση χρονικών διαρκειών**, εκτύπωση ως συμβολοσειρά, και η μέθοδος `abs()`

Σημαντικό:

- Αριθμητικές πράξεις (πρόσθεση, αφαίρεση) μεταξύ ημερομηνιών (αντικείμενα `date`) επιστρέφουν χρονικές διάρκειες (αντικείμενα `timedelta`).
- Μπορούμε να κάνουμε πράξεις μεταξύ `date` και `timedelta`
- Επίσης το ίδιο ισχύει για πράξεις μεταξύ ωρών (αντικείμενα `time`)

Παράδειγμα 6: timedelta/methods.py

```
from datetime import date, timedelta

d = date(2021, 12, 31)
t = timedelta(days=10)
print(d+t)

d2 = d-t
print(repr(d2-d))

print(date.today()+timedelta(days=10000))
```

Η κλάση datetime:

- Συνδυάζει τις κλάσεις date και time σε μία:

Κατασκευαστής	Επεξήγηση
date(year, month, day hour=0, minute=0, second=0, microsecond=0)	Αρχικοποιεί το αντικείμενο (οι πληροφορίες της ώρας είναι προαιρετικές)

- Όλα τα παραπάνω είναι μέλη της κλάσης
- Προσφέρονται οι μέθοδοι:
 - **isoformat()**, **fromisoformat(string)** Η μορφή της συμβολοσειράς είναι 'YYYY-DD-MMTTHH:MM:SS:μμμμμ' (το μέρος της ώρας είναι προαιρετικό)
 - Μπορούμε να κάνουμε **αριθμητικές πράξεις** με αντικείμενα timedelta, καθώς και **συγκρίσεις datetime**
 - **replace()**: με ορίσματα με λέξεις - κλειδιά.

Παράδειγμα 7: datetime/datetime_ops.py

```
from datetime import datetime, timedelta

d = datetime(2021, 11, 1, 13, 11, 39)
print(d.isoformat(sep=" "))
print(d, d + timedelta(hours=48))
print(d.replace(hour=23))

print(d < datetime(2022, 1, 1))
print(d == datetime(2021, 11, 1))
```

Επιπλέον Μέθοδοι:

Μέθοδος	Επεξήγηση
date()	Επιστρέφει αντικ/vo date
time()	Επιστρέφει αντικ/vo time
combine(date, time)	Συνδυάζει τα αντικ/να date, time σε ένα αντικείμενο datetime
today(), now(tz=None)	Επιστρέφει την τρέχουσα ημερομηνία/ώρα

Παράδειγμα 8: datetime/methods.py

```
# pip install pytz
from pytz import timezone
from datetime import datetime

now = datetime.now()
# tz names:
https://en.wikipedia.org/wiki/List\_of\_tz\_database\_time\_zones
us_now = datetime.now(timezone("America/New_York"))
kt_now = datetime.now(timezone("Asia/Katmandu"))

print("Greece time: " + str(now))
print("US/New York time: " + str(us_now))
print("Katmandu time: " + str(kt_now))
```

Μορφοποίηση Ημερομηνίας/Ωρας:

Μέθοδος	Επεξήγηση
strftime(string)	Επιστρέφει τη μορφοποίηση του αντικειμένου ως συμβολοσειρά που ακολουθεί τη μορφοποίηση string

- **string:** είναι συμβολοσειρά που περιέχει μορφοποιητές.

Μορφοποιητής	Επεξήγηση
%y	Διψήφιο έτος
%B	Πλήρες όνομα μήνα
%A	Πλήρες όνομα ημέρας
%I	Ώρα σε μορφή 12h
%p	AM ή PM
%M	zero-padded λεπτά
%S	zero-padded δευτερόλεπτα

- και υπάρχουν δεκάδες ακόμη μορφοποιητές:
(<https://docs.python.org/3/library/datetime.html?highlight=strftime#datetime.date.strftime>)

Παράδειγμα 9: format/strftime.py

```
from datetime import datetime
d = datetime.now()
print(d.strftime("%A %B %y"))
print(d.strftime("%I.%M.%S %p"))
```

- Οι κλάσεις date, time και datetime περιέχουν τη μαγική μέθοδο **format()** που έχει ακριβώς την ίδια λειτουργία με την strftime()

Μέθοδος	Επεξήγηση
__format__	Χρησιμοποιείται ως __format__(string) και είναι ίδια με τη strftime

- Επίσης μπορεί να γίνει αποδοτική ενσωμάτωση του **module locale**, το οποίο ενσωματώνει χαρακτηριστικά εντοπιότητας (όπως τα ονόματα των ημερών, μηνών κ.ο.κ.)

Παράδειγμα 10: format/locale_info.py

```
from datetime import datetime
import locale

for loc in ['greek', 'french', 'german', 'italian']:
    print("Language: " + loc)
    locale.setlocale(locale.LC_TIME, loc)
    d=datetime.now()
    for i in range(1,8):
        print(d.replace(day=i).__format__(f"%A %d %B %Y"))
    print("="*10)
```