



Czas i data w Pythonie

M@я3k Pүđ€£kø

Programowanie w Pythonie

Spis treści

- Python to skryptowy język interpretowalny o dużej liczbie bibliotek.
- Użyteczny do tworzenia frameworków sieciowych i innych operacji w Internecie.
- Posiada dużo wybudowanych narzędzi do przetwarzania zbiorów tekstowych. Zawiera wiele wbudowanych typów (listy, krotki, słowniki).
- Charakteryzuje się jasną i przejrzystą strukturą przy dużych możliwościach i elastyczności.

Czas komputerowy (uniksowy)

- Czas uniksowy, czas POSIX (ang. Unix time, POSIX time) to system reprezentacji czasu mierzący liczbę sekund od początku 1970 roku UTC, czyli od chwili zwanej początkiem epoki Uniksa (ang. Unix Epoch).
- Początek ery komputerowej to 1 stycznia 1970 godzina 0:00:00 czasu UTC.
- W systemie operacyjnym Unix i pochodnych czas jest przedstawiany jako 32-bitowa liczba sekund, które upłynęły od 1 stycznia 1970.
- Dostępny przedział czasu wynosi $2^{31}-1$ sekund, co daje wartość równą 2 147 483 647. 32-bitowy **Unix time** wyczerpie się 19 stycznia 2038 o godz. 03:14:07 UTC – wtedy pojawi się problem roku 2038.
- Przy 64-bitowej reprezentacji czasu (typ `time_t`) analogiczny problem pojawi się dopiero w roku 292 277 026 596, czyli za około 292 miliardy lat. 64-bitowa reprezentacja czasu zakończy się po upłynięciu $(2^{63}-1)$ sekund (9 223 372 036 854 775 807) od rozpoczęcia ery Uniksa.

Zalety czasu komputerowego

- Nie jest łatwo zarządzać i konwertować różne systemy czasu.
- Czas Unix oznacza, że każda data/czas przekłada się na jedną liczbę. Dzięki temu wszelkie operacje na czasie, dacie polegają na porównaniu dwóch liczb całkowitych.
- Dla komputera znacznie łatwiej jest śledzić jedną, bardzo dużą liczbę, która stale rośnie, niż ludzki system małych liczb sekund dodawanych do minut, godzin, dni itd.
- Komputer ma wbudowany system konwersji między wyświetlanym czasem a czasem Unix, więc późniejsza konwersja na lokalną datę/czas jest łatwa.
- wyświetlany czas JEST niezależny od UNIX-a. Ponieważ za każdym razem, gdy użytkownik chce wyświetlić czas, wykonywane jest obliczenie, które konwertuje czas UNIX na format czasu, który rozumie użytkownik.
- Znacznik czasu UNIX nie może reprezentować dat sprzed 1970 roku. Służy on tylko do reprezentowania bieżącego czasu systemowego. Od czasu wprowadzenia systemu UNIX w 1970 roku, daty sprzed tego czasu nigdy nie były istotne. To nie znaczy, że komputery nie mogą obsługiwać dat sprzed tego czasu, tylko że nie mogą być one reprezentowane jako standardowy znacznik czasu systemu UNIX.
- „Czas Unixowy” to po prostu system, który opracowali programiści Uniksa w latach 70. Stał się standardem de facto, a później standardem oficjalnym. Wiele różnych systemów musi być w stanie porównywać ze sobą znaczniki czasu, więc potrzebny jest spójny sposób na to. Znaczniki czasu Uniksa należą do najpopularniejszych.
- Jeśli piszesz wiadomości do współpracownika po drugiej stronie planety, jego „wyświetlany” czas jest zupełnie inny niż Twój, mimo że komunikujecie się w czasie rzeczywistym. Potrzebny jest spójny sposób reprezentowania czasu, niezależny od takich rzeczy jak strefy czasowe i czas letni.

Moduł time

- W języku python możliwość uzyskania informacji o czasie daje moduł **time**.

```
import time
```

```
print (time.time())
```

```
print (time.localtime())
```

Wyświetlanie czasu

- **time.ctime()**: Zwraca liczbę sekund (float) od początku epoki (1 stycznia 1970) czasu lokalnego.

```
print (time.ctime ( ) )
```

- **time.time()**: Zwraca liczbę sekund (float) od początku epoki (1 stycznia 1970) czasu UTC.

```
print (time.time ( ) )
```

Generowanie czasu z liczby

- **time.localtime():** Wyświetla datę w postaci krotki.

```
print (time.localtime())
```

```
#time.struct_time(tm_year=2026, tm_mon=1,  
tm_mday=23, tm_hour=13, tm_min=15, tm_sec=23,  
tm_wday=4, tm_yday=23, tm_isdst=0)
```

- Krotka ma specjalną strukturę.
 - Rok
 - Numer miesiąca
 - Dzień miesiąca
 - Godzina
 - Minuta
 - Sekunda
 - numer dnia tygodnia
 - numer dnia roku
 - oznaczenie czas letni czy zimowy (0 – czas zimowy, 1 – czas letni).

Generowanie czasu - pozycje

- Dostęp do poszczególnych pozycji krotki czasowej odbywa się poprzez indeks.

```
print(obj[0])          #rok  
print(obj[4])          #godzina
```

- Można również wykorzystać nazwy poszczególnych pól.

- **tm_year** Rok
- **tm_mon** Numer miesiąca
- **tm_mday** Dzień miesiąca
- **tm_hour** Godzina
- **tm_min** Minuta
- **tm_sec** Sekunda
- **tm_wday** numer dnia tygodnia
- **tm_yday** numer dnia roku
- **tm_isdst** oznaczenie czas letni / zimowy.

```
print(obj.tm_year)     #rok  
print(obj.tm_hour)     #godzina
```


Generowanie liczb z czasu

- **time.mktime(y)**: funkcja zwraca liczbę sekund dzielącą datę od punktu startowego czasu uniksowego (1 stycznia 1970 czasu UTC.).
- Argument **y** musi być napisaną krotką złożoną z 9 liczb.

```
import time
y=2001, 9, 9, 3, 46, 40, -1, -1, -1
print(time.mktime(y))          #1000000000
```

- Krotka, będąca argumentem funkcji **time.mktime()** musi mieć specjalną strukturę.
 - Rok (większy lub równy 1970)
 - Numer miesiąca (1..12).
 - Napisanie liczby ujemnej lub większej od 12 spowoduje korektę roku i zmianę na miesiąc odpowiedniego roku.
 - Dzień miesiąca (1..31).
 - Godzina (0..23)
 - Minuta (0..59)
 - Sekunda (1..61)
 - numer dnia tygodnia (0 ..6),
 - numer dnia roku (1..366)
 - Oznaczenie, czy jest to czas letni czy zimowy (0 – czas zimowy, 1 – czas letni).
- W miejsce nieznanych liczb można wpisać -1.
- Wpisanie liczby spoza przewidywanego zakresu generuje błąd.

Pauza

- **time.sleep(sekundy):** Wstrzymuje wykonanie programu.

```
print ("Hello")
```

```
time.sleep(2)
```

```
print ("my")
```

```
time.sleep(2)
```

```
print ("World")
```

- Ta opcja pozwala na stworzenie prostej animacji (zwłaszcza w połączeniu z czyszczeniem ekranu)

Moduł datetime

- W języku python możliwość uzyskania informacji o dacie i czasie daje moduł **datetime**.

```
import datetime  
print(datetime.datetime.now())
```

Przykład użycia daty i czasu

```
import datetime

# Aktualna data i czas
teraz = datetime.datetime.now()
#Format:YYYY-MM-DDHH:MM:SS.mmmmmm
print(teraz)

#Tylko aktualna data
dzis = datetime.date.today()
#Format:YYYY-MM-DD
print(dzis)

from datetime import datetime
teraz = datetime.now()
#Tylko aktualna godzina
print(f"Godzina: {teraz.time()}")
```

Ćwiczenia

1. Napisz program informujący o aktualnym:
 - a) czasie
 - b) dacie
2. Napisz program wczytujący dwie daty i obliczający ile sekund upłynęło między nimi.
3. Napisz program wyliczający za ile dni będzie weekend.
4. Napisz program, który pobierze dzisiejszą datę i policzy, ile dni zostało do końca bieżącego roku.
5. Program wczytuje imię i wyświetla je litera po literze, w odstępach co sekundę.
6. Wyszukaj jakiś kształt ASCII-ART i wyświetl go kolejno linijka po linijce w niewielkich odstępach czasowych.
7. Narysuj świerk rozwijający się od czubka w dół.

Powtórzenie

- Co to jest czas uniksowy?
- Dlaczego stosuje się czas uniksowy zamiast systemu rok-miesiąc-dzień-godzina-minuta-sekunda?
- Jakie możliwości daje moduł **time**?
- Jakie możliwości daje moduł **datetime**?
- Jak uzyskać prostą animację używając modułu **time**?