Järjendid (iterable)

Järjendid (ik *iterable*) on andmetüübid, mille väärtuste või elementide hulgast on võimalik eraldada osahulki (täisarvuliste) järjenumbrite ehk indeksi(te) abil. Järjenditeks on loend (ik *list*), ennik ehk korteež (ik *tuple*), ka tekst (string). Pythonis algavad järjendite indeksid alati nullist. Viitamisel järjendi elemendile/elemetidele paigutatakse indeks(id) järjendi nime järele nurksulgudesse.

s[0] – järjendi esimene element (teksti esimene sümbol või loendi esimene väärtus)
s[-1] – järjendi viimane element
s[2 : 5] – järjendi elemendid järjenumbritega 2, 3 ja 4 (ehk kolmandast viiendani)
s[1 : 9 : 2] – järjendi elemendid järjenumbritega 1, 3, 5 ja 7
s[:3], s[-3:] – kolm elementi järjendi algusest / lõpust

v ***in*** s, v **not in** s – väärtus on / ei ole järjendis (või hulgas)
len(s) – järjendi (või hulga) elementide arv
max(s), **min**(s) – suurim / vähim element järjendis (või hulgas)

Tekst (string)

String on mittemuudetav, st sümbolikaupa pole selles võimalik muudatusi teha (võib koostada teise stringi)

str.**count**(sub[,start [, end]]) – alamstringi esinemiste arv
str.**find**(sub[,start [, end]]) – alamstringi asukoht (index) , puudumisel –1; vt.ka *str*.rfind()
str.**index**(*sub*[, *start*[, *end*]]) – alamstringi asukoht (index), puudumisel viga ValueError
str.**join**(järjend) – järjendi elementidest (mis kõik peavad olema tekstid) moodustatakse tekst, milles eraldajaks on *str*
str.**split**([sep[, maxsplit]]) – jagab teksti osadeks, moodustades loendi; sep – eraldaja, võib koosneda mitmest sümbolist, vaikimisi tühik, tabulaator ja reavahetus
str.**strip**([chars]) – eemaldatakse antud sümbolid, vaikimisi tühikud, tabulaatorid, reavahetused
str.**lower**(); *str*.**upper**() – sümbolid väike- või suurtähtedeks
str.**isalnum**(), *str*.**isalpha**(), *str*.**isdecimal**(), *str*.**isdigit**(), *str*.**isnumeric**(), *str*.**islower**(), *str*.**isupper**()

Loend (list)

Loend on andmetüüp loetelu jaoks: muutujate järjestatud kogum, elementide poole pöördutakse **nimi[jnr]**, elemendid võivad olla erinevat tüüpi. Loendi väärtused ümbritsetakse nurksulgudega.

loend = [] – tühja loendi loomine
loend = **list**(....) – loendi loomine järjendist, hulgast jm
loend[i] = x – loendi i-s element asendatakse väärtusega x
loend[i:j] = t – loendi elemendid i:j asendatakse järjendi t elementidega
loend[i:j:k] = t – loendi elemente asendatakse elementidega teisest loendist, mille pikkus peab olema võrdne asendatavate elementide arvuga
del *loend*[i:j] – elemendid eemaldatakse loendist, ka del(loend[i:j])
del *loend*[i:j:k] – elemendid eemaldatakse loendist
loend.**append**(x) – lisab x–i (väärtus või objekt) loendi lõppu
loend.**extend**(L) – lisab antud loendi teise loendi lõppu
loend.**count**(x) – loendab x väärtuste arvu loendis
loend.**index**(x[, i[, j]]) – tagastab x–i asukoha (indeksi) loendis elementide vahemikus i:j
loend.**insert**(i, x) – lisab x loendisse järjenumbriga määratud elemendi ette
loend.**pop**([i]) – loeb viimase väärtuse, ühtlasi eemaldab selle loendist
loend.**remove**(x) – eemaldab loendist esimese x väärtuse
loend.**reverse**() – loendi elemendid vastupidises järjestuses
loend.**sort**() – sorteerib loendi elemendid kasvavas järjestuses (reverse=True – kahanevalt)

Fail (file)

Enne kasutamist tuleb fail avada

f = **open**(*fnimi*, *töötlusviis*)

f – objektimuutuja , f*nimi* – faili täisnimi: [***tee***]nimi.laiend , '**w**' – kirjutamine, '**r**' – lugemine (vaikimisi)

Pärast kasutamist tuleks fail sulgeda meetodiga close()

Failiobjekti omadusi ja meetodeid:

f.**name**	– faili nimi
f.**close**()	– faili sulgemine
f.**closed**	– kas fail on suletud
f.**readable**(), *f*.**writable**()	– kas fail on lugemiseks / kirjutamiseks
f.**read**()	– kõigi andmete lugemine, tulemuseks tekst
f.**readline**()	– ühe kirje (rea) lugemine
f.**readlines**()	– kogu teksti lugemine kirjete kaupa loendisse
f.**seek**()	– lugemisviit nihutatakse etteantud kohta
f.**seekable**()	– kas lugemisviita on võimalik nihutada
f.**tell**()	– lugemisviit – mitu sümbolit/baiti on loetud
f.**truncate**()	– faili tühjendamine
f.**write**()	– kirje (stringi) kirjutamine tekstifaili
f.**writelines**()	– loendi kirjutamine faili – ainult tekstid, elemendi lõpus soovitavalt '\n'

print(väärtus(ed), sep=' ', end='\n', file=*f_obj*) – tekstifaili kirjutamine (mugavam kui meetod write)

Sõnastik (dictionary)

Sõnastk on andmetüüp loetelu jaoks: elementide poole pöördutakse **nimi[võti]**, võti (ik *key*) võib olla peale täisarvu ka reaalarv, tekst, tõeväärtus. Sõnastiku moodustavad võti:väärtus paarid, mis eraldatakse komadega. Piirajatena on kasutusel looksulud {}. Sõnastik ei ole järjestatud.

d = {võti1:väärtus1, võti2:väärtus2, ...}; d = {} – tühja sõnastiku loomine

dict(...)	– sõnastiku loomine
len(d)	– elementide arv sõnastikus
d[võti] = väärtus	– uue elemendi loomine või vanale väärtuse omistamine
del d[võti]	– elemendi eemaldamine
võti **in** d, võti **not in** d	– kas selline võti on / ei ole sõnastikus
d.**clear**()	– sõnastiku tühjendamine
d.**pop**(võti)	– kui võti on sõnastikus, tagastab sellele vastava väärtuse ja eemaldab elemendi
d.**keys**()	– kõik võtmed sõnastikust
d.**values**()	– kõik väärtused sõnastikust