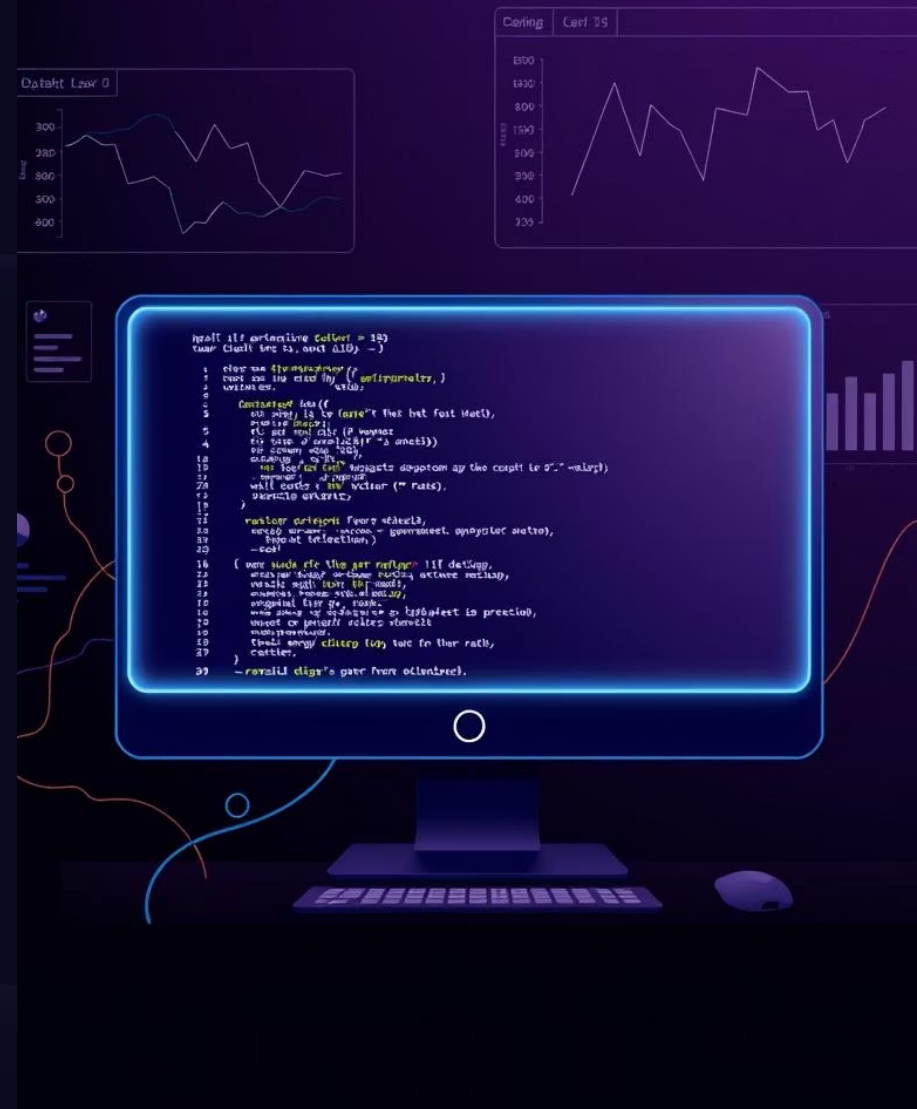


Дәріс 2. Деректерді талдауға арналған Python негіздері

Бүгінгі таңда 21-ғасырдың негізгі құндылықтарының бірі – деректер. Оларды тиімді жинау, өңдеу және түсіндіру мүмкіндігі заманауи ғылым мен экономика үшін өмірлік маңызды салаға айналды. Осы орайда, Python деректерді талдау және ғылыми зерттеу саласында қолданылатын әмбебап құрал ретінде ерекше орын алады.

Оқытушы: Сарсембаева Т.С.



Дәрістің мақсаты

Python тілінің негізгі құрылымдарын, синтаксистік ережелерін, деректер түрлерін және ағынды басқаруды меңгерту. Сонымен қатар, Python-ды деректерді талдау, машиналық оқыту және жасанды интеллект салаларындағы негізгі құрал ретінде түсіндіру, сондай-ақ осы тілді зерттеу барысында зерттеушінің алгоритмдік, логикалық және ғылыми-зерттеушілік ойлау дағдыларын дамыту көзделеді.

Дәріс соңында Python-ның әмбебаптығы және оның биоинформатикадан эконометрикаға дейінгі әртүрлі ғылыми салаларда қолданылуы көрсетіледі.

Синтаксис негіздері

Оқуға оңай, интуитивті құрылым

Деректер түрлері

str, list, dict, set құрылымдары

Ағынды басқару

if, for, while операторлары

Негізгі сұрақтар

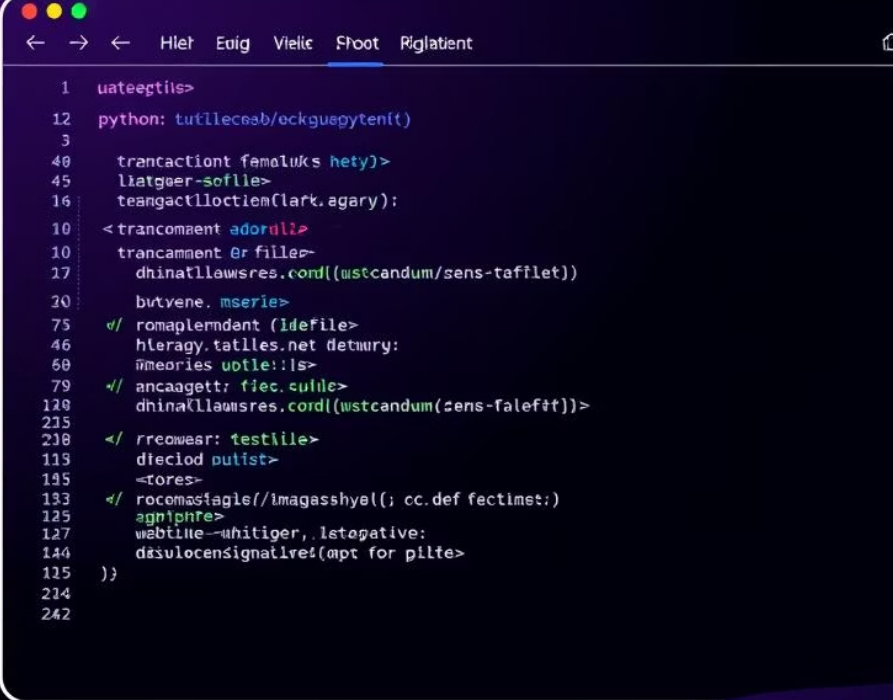
- 1 Python синтаксисінің негіздері қандай?
- 2 Python-дағы негізгі деректер түрлері қандай?
- 3 Ағынды басқару және функциялар қалай құрылады?
- 4 Объектіге-бағытталған бағдарламалау (ОББ) дегеніміз не және оның негізгі элементтері қандай?
- 5 Генераторлар, `zip()`, `args/kwargs` және тип аннотациялары не үшін қажет?

Синтаксис негіздері

Python синтаксисі – бұл Python тілінің грамматикалық және құрылымдық негізі, яғни бағдарламаның қалай жазылатынын және түсіндіретінін анықтайтын ережелер жиынтығы. Синтаксис бағдарламашы мен компьютер арасындағы "ортақ тіл" іспеттес: ол кодтың дұрыс жазылуын және орындалуын қамтамасыз етеді.

Python-ның ең көрнекті ерекшелігі – оның оқуға оңай және интуитивті синтаксисі. Python тілін оқу барысында бағдарламашы бағдарламалаудың негізгі дағдыларымен қатар, алгоритмдік және логикалық ойлау дағдыларын да меңгереді. Бағдарламалау процесінде қолданушы жүйелік түрде мәселенің құрылымын, деректер арасындағы қарым-қатынасты талдап, оны шешудің ең тиімді әдісін іздейді.

Бұл процесс ғылыми және зерттеушілік ойлауды дамытып, нақты деректер бойынша негізделген қорытындылар жасау қабілетін арттырады. Сондай-ақ, Python бағдарламашыларға модельдерді визуализациялауға, деректерді өңдеуге, есептеулер жүргізуге және қорытынды жасауға мүмкіндік береді.



```
1 uateegtiis>
12 python: tutilec6ab/ockguspytenft)
3
48 transaction femaluks hety>
45 llatger-sotlie>
16 teangactlloctiem(lark.agary):
10 <trancomment adordille>
10 trancomment 0r filler>
17 dhinatllawsres.cordl(wstcandum/sens-taffilet))
30 butvene. mserie>
75 </ romaplemdant (idefile>
46 hleragy.tatlles.net detmury:
60 meories uotle:ils>
79 </ ancaagett: ftec.culile>
128 dhinatllawsres.cordl(wstcandum(sens-falefft))>
215
218 </ rreowasr: testtile>
113 dieclod outist>
125 <tores>
133 </ rocomastagle//lmagasshyel(; cc.def fectlmst:)
125 agniphre>
127 webille--whitiger,.lstapative:
144 dsulocensignaltres(mpt for pilte>
125 ))
214
242
```

Негізгі операторлар

Python тіліндегі операторлар — айнымалылармен және мәндермен әртүрлі әрекеттерді орындауға арналған таңбалар немесе кілт сөздер болып табылады. Олар бағдарламалауда логикалық, арифметикалық және басқару амалдарын орындауға мүмкіндік береді.

Арифметикалық операторлар

Математикалық есептеулер жүргізу үшін қолданылады: +, -, *, /, //, %, **

Салыстыру операторлары

Екі мәнді салыстыру үшін қолданылады:
==, !=, >, <, >=, <=

Логикалық операторлар

Бірнеше шарттарды біріктіру үшін қолданылады: and, or, not

Мысалдар:

```
x = 10
y = 3
print(x + y) # 13
print(x - y) # 7
print(x * y) # 30
print(x / y) # 3.333...
print(x // y) # 3
print(x % y) # 1
print(x ** y) # 1000
```

Блоктар, шегіністер және деректер түрлері

Python тілінің тағы бір маңызды синтаксистік ерекшелігі - блоктар мен шегіністерді қолдану ережесі. Басқа көптеген тілдерде – C++, Java, JavaScript тілдерінде блоктар {} жақшалар арқылы жазылса, Python-да кодтың құрылымы шегініс деңгейімен көрсетіледі. Блок – белгілі бір логикалық бірлікті білдіретін код тобы.

Деректер түрлері Python-ның негізгі ерекшеліктерінің бірі болып табылады. Деректерді дұрыс қолдану арқылы, кейінгі талдаудың дәлдігі мен тиімділігі қамтамасыз етіледі.



Ағынды басқару және функциялар

Python бағдарламалау тілінде басқару ағыны (control flow) және функциялар (functions) - бағдарламаның логикалық құрылымын, модульдік ұйымдастырылуын және сенімділігін қамтамасыз ететін негізгі ұғымдар болып табылады.



Шартты операторлар

if, elif, else кілт сөздерін пайдаланып, логикалық жағдайға байланысты әртүрлі тармақтарды орындау



Циклдік құрылымдар

for және while циклдері арқылы қайталанатын әрекеттерді автоматтандыру



Функциялар

def арқылы анықталатын, қайта пайдалануға болатын код блогы

Функциялардың артықшылықтары: Кодтың қайталануын азайтады (DRY), модульділікті қамтамасыз етеді, тестілеуді жеңілдетеді, жадты үнемдейді.

Python

```
pete i ictch (1, 8)
starr Watch: 11, 4)
bu::lowed on parteting (1, 6, 1)
1   reccir: 1,21:
0   sâret woor: 2
> case the larur eloges 2,1:
```

ОББ, генераторлар және аннотациялар

Объектіге бағытталған бағдарламалау (Object-Oriented Programming) – бұл объектілер мен сыныптарды қолдана отырып бағдарлама құру тәсілі. Python – толыққанды ОББ тілдерінің бірі.

ОББ негіздері

- Класс – объектінің моделі
- Объект – класс негізінде жасалған нақты мысал
- Атрибуттар – объектінің деректері
- Әдістер – объект орындайтын әрекеттер

Генераторлар және аннотациялар

Генераторлар: yield арқылы деректерді ретімен береді, жадты үнемдейді.

zip(): Бірнеше тізімді жұптап біріктіреді.

args/kwargs: Белгісіз санды аргументтерді қабылдайды.

Аннотациялар: Кодтың оқылуын жақсартады, күтілетін деректер түрін көрсетеді.

Python негіздерін меңгеру – синтаксис, мәліметтер типтері, ағындарды басқару, функциялар, Объектіге бағытталған бағдарламалау принциптері – деректерді талдау процесінің негізі болып табылады. Сонымен қатар, аннотациялар, генераторлар және args/kwargs сияқты құралдар бағдарламаның құрылымын анық және сенімді етуге көмектеседі.

Әдебиеттер тізімі

1. Fractal Data Science Professional Certificate (соның ішінде "Python for Data Science" курсы) — <https://www.coursera.org/professional-certificates/fractal-data-science> Coursera
2. Python For Data Analytics: A Systematic Literature Review of Tools, Techniques, and Applications - https://papers.ssrn.com/sol3/papers.cfm?abstract_id=5051680
3. McKinney, W. (2021). Python for Data Analysis. In Python Data Analytics. Springer, Singapore. DOI: https://doi.org/10.1007/978-981-16-8615-3_7