შეხვედრა 10: მონაცემთა ანალიზის ბიბლიოთეკა : Pandas

აქამდე ჩვენი ჩატბოტისთვის მონაცემებს თავად ვქმნიდით და თავად ვუწერდით ამ მონაცემების მართვასთან დაკავშირებულ წესებს. ასეთი მიდგომა პატარა პროექტებისთვის გამართლებულია. მაგრამ რა ხდება, როცა მონაცემთა ბაზა მოიცავს რამდენიმე ათას ან მილიონ ერთეულს? ასეთი მონაცემების ხელით დამუშავება წარმოუდგენლად რთულია. სწორედ ამიტომ, დღეს ჩვენ გავეცნობით მონაცემთა ანალიზის ერთ-ერთ ყველაზე პოპულარულ და ძლიერ ინსტრუმენტს - **Pandas ბიბლიოთეკას**. ამ შეხვედრის ბოლოს შენ შეძლებ, დაწერო პროგრამა, რომელიც დაამუშავებს დიდი მოცულობის მონაცემებს, დაათვალიერებს მათ სტრუქტურას და დაითვლის მონაცემებში არსებული კატეგორიების რაოდენობას. ეს არის ფუნდამენტური უნარი, რომელიც დაგჭირდება ხელოვნურ ინტელექტთან მუშაობისას, ნებისმიერ, თუნდაც ყველაზე რთულ პროექტში.

1. რა არის ბიბლიოთეკა პროგრამირებაში?

ბიბლიოთეკა არის მზა კოდის ერთობლიობა, რომელიც სხვა პროგრამისტებმა შექმნეს. მისი გამოყენებით, ჩვენ აღარ გვიწევს ბევრი და რთული კოდის თავიდან დაწერა. ბიბლიოთეკა შეგიძლია წარმოიდგინო, როგორც ერთგვარი ინსტრუმენტების ნაკრები, რომლის გამოყენებაც შენს პროექტებში შეგიძლია.

1.1. მზა ინსტრუმენტების გამოყენების უპირატესობა

წარმოიდგინე, რომ შენი ლეკვისათვის სახლის აშენება გინდა. შეგიძლია, ყველა საჭირო ინსტრუმენტი (ჩაქუჩი, ხერხი, ლურსმანი) თავად დაამზადო, ან შეგიძლია, უბრალოდ წახვიდე მაღაზიაში და იყიდო მზა ნაკრები. პროგრამირებაში ბიბლიოთეკა არის ასეთი მზა „ინსტრუმენტების ყუთი“, რომელიც სხვა პროგრამისტებმა უკვე შექმნეს და გამოსცადეს. Pandas ბიბლიოთეკა შეიცავს ყველა იმ ინსტრუმენტს, რომელიც მონაცემების გასაანალიზებლად და დასამუშავებლად დაგჭირდება.

1.2. `import pandas as pd`: როგორ "გამოვიძახოთ" პროფესიონალი მონაცემთა ანალიტიკოსი

იმისათვის, რომ Pandas-ის ინსტრუმენტები გამოვიყენოთ, ის ჩვენს პროგრამაში უნდა „შემოვიტანოთ“ `import` ბრძანების გამოყენებით.

```
# Pandas ბიბლიოთეკის შემოტანა
import pandas as pd
```

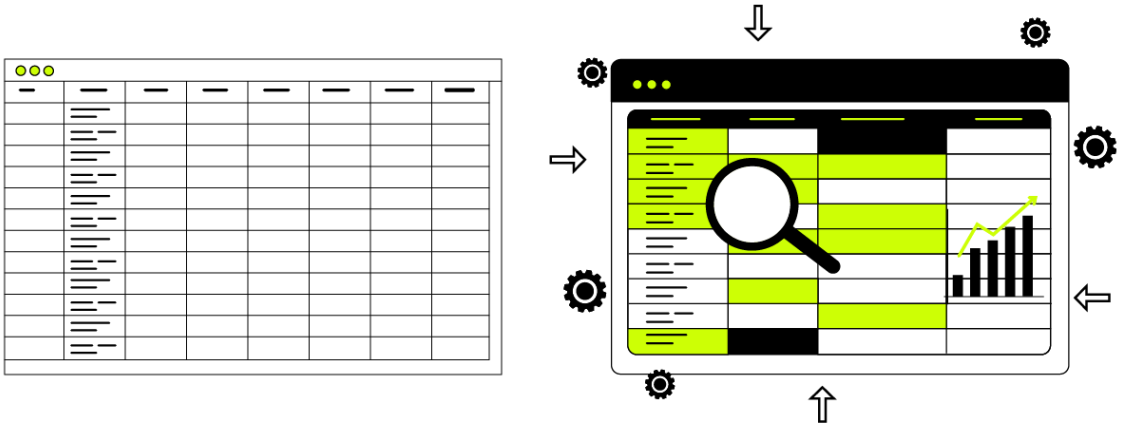
ბრძანება `import pandas as pd` ნიშნავს, რომ ჩვენ „შემოგვაქვს“ Pandas ბიბლიოთეკა და ვანიჭებთ მას მოკლე სახელს `pd`. ამიერიდან, ყოველ ჯერზე, როდესაც Pandas-ის ფუნქციის გამოძახება მოგვინდება, უბრალოდ დავწერთ `pd.` ეს ჰგავს ზედმეტსახელს, რომელიც აადვილებს ბიბლიოთეკასთან მუშაობას.

2. Pandas ბიბლიოთეკა

Pandas-ი მძლავრი ინსტრუმენტია, რადგან მონაცემებთან მუშაობას ბევრად ამარტივებს. მისი მთავარი მიზანია, მონაცემებთან მუშაობა იყოს მარტივი, ინტუიციური და სწრაფი. Pandas-ის შექმნის იდეა 2008 წელს გაჩნდა, როდესაც პროგრამისტი უეს მაკ-კინი (Wes McKinney) ფინანსური მონაცემების ანალიზს ცდილობდა, მაგრამ შესაბამის ინსტრუმენტს ვერ პოულობდა . სწორედ ამიტომ, მან გადაწყვიტა, თავად შეექმნა ასეთი ბიბლიოთეკა. ასე შეიქმნა Pandas-ი, რომელიც დღეს მონაცემთა მეცნიერების ერთ-ერთი ფუნდამენტური ინსტრუმენტია.

2.1. DataFrame - მონაცემების წარმოდგენა ჭკვიანი ცხრილის სახით

Pandas-ის მთავარი სტრუქტურული ელემენტი არის **DataFrame**, რომელიც მონაცემებს ინახავს ჭკვიანი ცხრილის სახით. ის ჰგავს Excel-ის ან Google Sheets-ის ცხრილს, რომელსაც ახასიათებს მწკრივები და სვეტები, მაგრამ ბევრად უფრო მეტი შესაძლებლობა აქვს მონაცემების გასაანალიზებლად.



2.2. მონაცემების ჩატვირთვა CSV ფაილის სახით

მონაცემები ხშირად ინახება .csv (Comma-Separated Values) ფორმატის ფაილებში. ეს არის ტექსტური ფაილი, სადაც მონაცემები ერთმანეთისაგან მძიმით არის გამოყოფილი. Pandas-ს შეუძლია მარტივად ჩატვირთოს ასეთი ფაილები read_csv() ფუნქციის გამოყენებით.

```
# pandas ბიბლიოთეკის შემოტანა
import pandas as pd
import io

# მონაცემების სიმულაცია CSV ფორმატში
data = """text,intent
გამარჯობა, greeting
როგორ ხარ, question
ნახვამდის, goodbye
ვინ ხარ, question
შენი სახელი რა არის, question
მოგესალმები, greeting
კარგად, goodbye
"""

# მონაცემების DataFrame-ში ჩატვირთვა
df = pd.read_csv(io.StringIO(data))

# მონაცემების პირველი 5 მწკრივის ჩვენება
print(df.head())
```

2.3. მონაცემების დათვალიერება: `head()`, `describe()`, `info()`

როდესაც მონაცემებს ვტვირთავთ, პირველ რიგში, ისინი უნდა დავათვალიეროთ, რათა გავიგოთ, რასთან გვაქვს საქმე.

- **`.head()`:** აჩვენებს ცხრილის პირველ ხუთ მწკრივს. ეს ძალიან მოსახერხებელია მონაცემების სტრუქტურის სწრაფად შესამოწმებლად.
- **`.info()`:** აჩვენებს ცხრილის ზოგად ინფორმაციას: სვეტების სახელებს, მონაცემთა ტიპებს და ცარიელი უჯრების რაოდენობას.
- **`.describe()`:** აჩვენებს რიცხვითი სვეტების სტატისტიკურ შეჯამებას, როგორიცაა საშუალო, მინიმალური, მაქსიმალური მნიშვნელობა და ა.შ.

ინტერაქტიული სავარჯიშო 1:

გამოიყენე ფუნქცია, რათა დაათვალიერო ჩვენი მონაცემთა ბაზის პირველი სამი რიგი.

```
import pandas as pd

import io


data = """text,intent
```

```
გამარჯობა, greeting

როგორ ხარ, question

ნახვამდის, goodbye

ვინ ხარ, question

შენი სახელი რა არის, question

მოგესალმები, greeting

კარგად, goodbye
"""


df = pd.read_csv(io.StringIO(data))


# დაწერე შენი კოდი აქ: გამოიყენე head() ფუნქცია, რომ დაბეჭდო ცხრილის პირველი 3 რიგი.

# ფუნქციის შიგნით მიუთითე რიცხვ 3-ს.

print(df.head(3))
```

3. მონაცემებთან მუშაობა

Pandas-ი გვაძლევს საშუალებას, მარტივად ვიმუშაოთ ცხრილებთან.

3.1. კონკრეტულ სვეტებსა და მწკრივებზე წვდომა

იმისათვის, რომ კონკრეტულ სვეტს მივწვდეთ, ვიყენებთ კვადრატულ ფრჩხილებს.

```
# მხოლოდ "text" სვეტის შერჩევა და პირველი 5 ელემენტის დაბეჭდვა
print(df['text'].head())
```

3.2. მონაცემების ფილტრაცია და საბაზისო ანალიზი

Pandas-ი საშუალებას გვაძლევს, მარტივად დავათვალიეროთ მონაცემები. მაგალითად, შეგვიძლია დავთვალოთ, თითოეული intent (განზრახვა) რამდენჯერ გვხვდება. ამისთვის ვიყენებთ value_counts() მეთოდს.

```
# დაითვალე თითოეული განზრახვის რაოდენობა
print(df['intent'].value_counts())
```

დავალება 10: მონაცემების დათვალიერება

გამოიყენე Pandas-ი, რათა ჩატვირთო CSV ფაილი, რომელიც შეიცავს 2 სვეტს: text და intent. დაბეჭდე: 1. მონაცემთა ბაზის პირველი 5 რიგი. 2. თითოეული განზრახვის (intent) რაოდენობა მონაცემთა ბაზაში.

დავალება 10.1: შეცდომის პოვნა და გასწორება

მოცემულ კოდში დაშვებულია შეცდომა. შენი ამოცანაა, იპოვო და გაასწორო ის, რათა პროგრამამ პირობის შესაბამისად იმუშაოს.

```
import pandas as pd
import io

data = """text,intent
გამარჯობა, greeting
როგორ ხარ, question
ნახვამდის, goodbye
ვინ ხარ, question
შენი სახელი რა არის, question
მოგესალმები, greeting
კარგად, goodbye
"""

# შეცდომით დაწერილი კოდი
df pd.read_csv(io.StringIO(data))
print(df.head)
```

დავალება 10.2: კოდის დასრულება

მოცემულ პროგრამულ კოდს აკლია ერთი ან რამდენიმე სტრიქონი. დაამატე მხოლოდ ის, რაც აუცილებელია, რომ პროგრამამ გამართულად იმუშაოს.

```
import pandas as pd
import io

data = """text,intent
გამარჯობა, greeting
როგორ ხარ, question
ნახვამდის, goodbye
ვინ ხარ, question
შენი სახელი რა არის, question
მოგესალმები, greeting
კარგად, goodbye
"""

df = pd.read_csv(io.StringIO(data))

# შენი კოდი აქ, რომელიც დაითვლის განზრახვების რაოდენობას
```

დავალება 10.3: კოდის დაწერა ნულიდან
დაწერე პროგრამული კოდი, შექმენი პროგრამა, რომელიც შექმნის Pandas DataFrame-ს მოცემული მონაცემებიდან, დაბეჭდავს მის პირველ 5 მწკრივს და დაითვლის განზრახვების (intent) რაოდენობას.

```
# დაწერე შენი კოდი აქ:
```

სწორი პასუხი (პროგრამული კოდი სრულად):

```
# ამ კოდის საშუალებით შეგიძლია გადაამოწმო შენი ნამუშევარი

# 1. საჭირო ბიბლიოთეკების შემოტანა
import pandas as pd
import io

# 2. მონაცემების სიმულაცია
data = """text,intent
გამარჯობა, greeting
როგორ ხარ, question
ნახვამდის, goodbye
ვინ ხარ, question
შენი სახელი რა არის, question
```

```
მოგესალმები, greeting
კარგად, goodbye
"""

# 3. მონაცემების DataFrame-ში ჩატვირთვა
df = pd.read_csv(io.StringIO(data))

# 4. მონაცემების პირველი 5 მწკრივის დაბეჭდვა
print("მონაცემთა ბაზის პირველი 5 რიგი: ")
print(df.head())

# 5. თითოეული განზრახვის რაოდენობის დათვლა და დაბეჭდვა
print("\nთითოეული განზრახვის რაოდენობა:")
print(df['intent'].value_counts())
```