№7 ЗЕРТХАНАЛЫҚ ЖҰМЫС
NumPy құралдарымен өлшеу массивтерін сандық өңдеу

1. Жұмыстың мақсаты
NumPy кітапханасының көмегімен көлік жүйелерінің өлшеу деректерін массивтерге сақтау, статистикалық есептеулер жүргізу, математикалық операциялар орындау және өлшеу мәндерін берілген шарттар бойынша талдау.

2. Теориялық мәліметтер
NumPy (Numerical Python) — Python тілінде сандық есептеулерді орындауға арналған кітапхана.

NumPy кітапханасының негізгі құрылымы — ndarray, яғни көпөлшемді массив.

Мысалы, Python тізімі:
temperature = [88, 92, 95, 101, 106]

NumPy массиві:

```
import numpy as np

temperature = np.array([88, 92, 95, 101, 106])
```

Екі құрылым да бірнеше мәнді сақтай алады. Бірақ NumPy массивтері біртекті сандық деректермен математикалық операцияларды тиімді орындауға арналған.

Python тізімі	**NumPy массиві**
list	ndarray
Әртүрлі типтегі элементтерді сақтай алады	Әдетте бір деректер типіндегі элементтерден тұрады
Математикалық операциялар көбінесе цикл арқылы орындалады	Массив элементтеріне операцияны бірден қолдануға болады
sum(), min(), max()	np.sum(), np.min(), np.max(), np.mean()

3. Жұмыс ортасын дайындау
Google Colab платформасында жаңа notebook құрыңыз.
NumPy кітапханасын импорттаңыз:

import numpy as np

Google Colab ортасында NumPy әдетте алдын ала орнатылған.

Тапсырма №1. NumPy массивін құру
Электрмобиль батареясының температурасы 10 рет өлшенді.
Бастапқы деректер:
temperature = [28, 30, 32, 35, 38, 42, 45, 48, 52, 55]

Осы мәндерден NumPy массивін құрыңыз.

```
import numpy as np

temperature = np.array([28, 30, 32, 35, 38, 42, 45, 48, 52, 55])

print("Temperature:", temperature)
print("Number of measurements:", temperature.size)
print("Data type:", temperature.dtype)
```

Орындау қажет: массив элементтерінің санын, деректер типін және массив өлшемін (shape) анықтау.

Тапсырма №2. Статистикалық көрсеткіштерді есептеу
№1 тапсырмадағы массивті пайдаланып, келесі көрсеткіштерді есептеңіз:

1. Минималды температура.
2. Максималды температура.
3. Орташа температура.
4. Медиана.
5. Стандартты ауытқу.

```
print("Minimum:", np.min(temperature))
print("Maximum:", np.max(temperature))
print("Average:", np.mean(temperature))
print("Median:", np.median(temperature))
print("Standard deviation:", np.std(temperature))
```

Тапсырма №3. Массив элементтеріне математикалық операциялар қолдану

Температура датчигінің өлшеу нәтижелерінде тұрақты +2°C қате бар деп есептейік.
Барлық өлшеулерді 2°C-қа азайту қажет.

```
corrected_temperature = temperature - 2

print("Original:", temperature)
print("Corrected:", corrected_temperature)
```

Неліктен NumPy массивімен мұндай операцияны for циклінсіз орындауға болатынын түсіндіріңіз

Тапсырма №4. Массив элементтерін шарт бойынша сүзу

Температура	**Күйі**
T < 15°C	LOW
15°C ≤ T ≤ 45°C	NORMAL
T > 45°C	HIGH

Температурасы 45°C-тан жоғары өлшеулерді табыңыз.

```
import numpy as np
temperature = np.array([28, 30, 32, 35, 38, 42, 45, 48, 52, 55])
high_temperature = temperature[temperature > 45]

print("High temperatures:", high_temperature)
print("Count:", high_temperature.size)
```

```
High temperatures: [48 52 55]
Count: 3
```

Қосымша тапсырмалар:

1. 15°C-тан төмен мәндерді табу.
2. Қалыпты диапазондағы мәндерді табу.
3. Әр санаттағы өлшеулер санын есептеу.
4. Қалыпты диапазоннан тыс барлық мәндерді анықтау.

сүзу (Filtering) — деректердің ішінен белгілі бір шартқа сәйкес келетін мәндерді ғана таңдап алу.

Қалыпты диапазонды сүзу мысалы:

```
normal = temperature[(temperature >= 15) & (temperature <= 45)]

print("Normal temperatures:", normal)
```

NumPy массивтеріндегі шарттарды біріктіру үшін & (AND) және | (OR) операторлары қолданылады. Әр шарт жақшаға алынады.

Тапсырма №5. Өлшеу деректерін нормализациялау

Машиналық оқытуда әртүрлі диапазондағы өлшеулерді бір масштабқа келтіру жиі қолданылады.
Min-Max нормализациясы мәндерді 0 мен 1 аралығына түрлендіреді.
Формула:

$$x_{\text{norm}} = \frac{x - x_{\min}}{x_{\max} - x_{\min}}$$

Python бағдарламасы:

```
minimum = np.min(temperature)
maximum = np.max(temperature)

normalized = (temperature - minimum) / (maximum - minimum)

print("Normalized:", normalized)
```

Тапсырма №6. Екі массивпен жұмыс істеу

Электрмобиль батареясының кернеуі мен тогы өлшенді.

```
voltage = np.array([380, 378, 375, 372, 370])
current = np.array([20, 25, 30, 35, 40])
```

Әр өлшеу үшін электр қуатын есептеңіз.
Формула:

$P = U \cdot I$
Мұнда P — қуат (Вт), U — кернеу (В), I — ток (А).

```
power = voltage * current

print("Power (W):", power)
print("Maximum power:", np.max(power))
print("Average power:", np.mean(power))
```

Жеке тапсырмалар

Әр білім алушы өз нұсқасына сәйкес екі параметрден тұратын NumPy массивтерін құрып, сандық есептеулер мен деректерді сүзу операцияларын орындауы тиіс.

№	Тақырып	Жеке тапсырма
1	Электрмобильдің батарея кернеуі мен зарядтау тогы	Кернеуі 400 В-тан жоғары және зарядтау тогы 50 А-дан жоғары болатын өлшеулерді анықтау, олардың санын есептеу
2	Жылдамдық пен уақыт	Әр уақыт аралығындағы жүрген қашықтықты есептеу
3	Батарея температурасы мен кернеуі	Екі параметрдің қалыпты диапазоннан ауытқуын анықтау
4	Қозғалтқыш RPM және температурасы	Екі параметр бойынша қауіпті өлшеулерді табу
5	Шина қысымы мен температурасы	Екі параметрді бір уақытта шарт бойынша сүзу
6	Кернеу мен ток	Қуатты есептеп, ең үлкен қуатқа сәйкес өлшеуді табу
7	SOC және батарея температурасы	Заряд төмен және температура жоғары болатын өлшеулерді табу
8	Жылдамдық пен отын шығыны	Екі параметрдің орташа мәнін және ауытқуларын салыстыру
9	Қозғалтқыш және салқындатқыш температуралары	Әр өлшеудегі температура айырмасын есептеу
10	Екі температура датчигі	Датчиктер көрсеткіштерінің айырмасын есептеп, айырмасы 3°C-тан жоғары өлшеулерді табу

Барлық нұсқаларға ортақ талаптар

1. NumPy кітапханасын импорттау.
2. Әр параметр үшін кемінде 10 өлшеуден тұратын массив құру.
3. Әр массивтің минималды, максималды және орташа мәндерін есептеу.
4. Нұсқаға сәйкес математикалық операцияларды орындау.
5. Берілген шарттар бойынша массивтерді сүзу.
6. Шарттарға сәйкес келетін өлшеулер санын анықтау.
7. Нәтижелерді экранға шығару.