№9 ЗЕРТХАНАЛЫҚ ЖҰМЫС
Деректер жиынтығын тазалау және алдын ала дайындау

Құралдар: Google Colab, Python, Pandas
Ұзақтығы: 2 академиялық сағат

1. Жұмыстың мақсаты
Дайын CSV файлындағы қайталанатын жолдарды, қате мәтіндік жазбаларды және бос ұяшықтарды анықтау, Pandas көмегімен өңдеу және тазаланған нәтижені сақтау.

2. Жұмыста не істейміз?
Оқытушы әр студентке көлік жүйесінің өлшеулері жазылған бір CSV файлын береді.
Файлда әдейі енгізілген мәселелер бар:

Мәселе	Мысал	Орындалатын әрекет
Қайталанатын жазба	Бір өлшеу бірнеше рет сақталған	Артық көшірмені жою
Сандық бағандағы мәтін	unknown, error	NaN мәніне ауыстыру
Бос ұяшық	Өлшеу мәні көрсетілмеген	Тиісті бағанның медианасымен толтыру

3. Негізгі ұғымдар
DataFrame — Pandas кітапханасындағы жолдар мен бағандардан тұратын кесте.
Осы жұмыста екі кесте қолданамыз:

raw	Файлдан оқылған бастапқы кесте
df	Тазалау әрекеттері орындалатын кесте

ID — өлшеудің идентификаторы, яғни оны басқа өлшеулерден ажырататын нөмір. ID өлшенген шама емес. Сондықтан оның медианасын есептемейміз. ID бағаны кестеде сақталады.

Индекс — Pandas кестесінің сол жағында көрсетілетін жол белгісі. CSV файлын әдеттегі тәсілмен оқығанда индекс 0-ден басталады. Индекс пен ID — екі бөлек нәрсе.

NaN — бос немесе жоқ сандық мәннің белгісі.

Дубликат — барлық бағандарының мәндері басқа жолмен толық сәйкес келетін жазба. Осы жұмыста салыстыруға ID бағаны да кіреді, ал индекс кірмейді.

Медиана — сандарды өсу ретімен орналастырғандағы ортаңғы мән. Сандар саны жұп болса, ортадағы екі санның арифметикалық ортасы алынады.

4. Бастапқы файл
Variant_01.csv – variant_10.csv файлдарының өз нұсқаңызға сәйкес біреуін алыңыз. Әр файлда тақырып жолынан бөлек 30 дерек жолы бар. Деректерді өзіңіз құру немесе бастапқы CSV файлын қолмен түзету қажет емес.

5. Файлды оқу және тексеру
Google Colab ортасында Lab9_Surname notebook құрыңыз. «Файлдар» панеліндегі тізімді жоғары қарай жылжытып, content бумасын табыңыз. Буманы ашу үшін content атауының сол жағындағы үшбұрышты басыңыз. content бумасын тінтуірдің оң жақ батырмасымен басып, «Жүктеу» / Upload пәрменін таңдаңыз. Өз variant.csv файлын таңдаңыз.

6. CSV файлын оқу
Жаңа ұяшыққа жазыңыз:

```
import pandas as pd
# Өз нұсқаңыздың файлын көрсетіңіз
filename = "variant_01.csv"
# CSV файлын бастапқы кестеге оқимыз
raw = pd.read_csv(filename)
# Бастапқы кестенің барлық жолын көрсетеміз
print("Бастапқы кесте:")
print(raw.to_string(index=True))
```

7. Кестенің құрылымын зерттеу
Жаңа ұяшыққа жазыңыз:

```
print("Алғашқы 5 жол:")
print(raw.head())

print("\nЖолдар саны:")
print(raw.shape[0])
```

```
print("\nБағандар саны:")
print(raw.shape[1])

print("\nБаған атаулары:")
print(raw.columns.tolist())

print("\nӘр бағанның деректер типі:")
print(raw.dtypes)

print("\nКесте туралы жалпы ақпарат:")
raw.info()

# Бастапқы жолдар санын сақтаймыз
rows_before = len(raw)
```

8. Қайталанатын жолдарды анықтау
Жаңа ұяшыққа жазыңыз:

```
# Әр жолдың артық көшірме екенін тексереміз
duplicate_check = raw.duplicated()

# Тек артық көшірмелерді таңдаймыз
duplicate_rows = raw[duplicate_check]

# Артық көшірмелердің санын есептейміз
duplicate_count = int(duplicate_check.sum())

print("Қайталанатын жолдардың артық көшірмелері:")
print(duplicate_rows)

print("\nАртық көшірмелер саны:")
print(duplicate_count)
```

Код қалай жұмыс істейді?

duplicate_mask = raw.duplicated() – Бұл команда әр жолды алдындағы жолдармен салыстырып, оның қайталанатынын тексереді.

raw.duplicated() әр жолға белгі қояды:

- False — осы жазба алғаш рет кездесті;
- True — осы жазба бұрын кездескен, бұл артық көшірме.

raw[duplicate_check] тек True белгісі бар жолдарды таңдайды.
sum() осы белгілерді санайды, себебі есептеу кезінде True — 1, ал False — 0 ретінде қабылданады.

Мысалы, бірдей жол үш рет кездессе, бірінші данасы сақталатын негізгі жазба, ал қалған екеуі артық көшірме болып саналады.

duplicate_mask — осы True және False белгілерін сақтайтын айнымалының атауы. Мысалы, raw кестесі:

Индекс	ID	Temperature
0	1	30
1	2	35
2	1	30

raw.duplicated() нәтижесі:
0-жол → False: бұл жазба алғаш рет кездесті.
1-жол → False: бұл да жаңа жазба.
2-жол → True: барлық мәндері 0-жолмен бірдей.

9. Дубликат/Қайталанатын жолдарды жою

```
# Бірінші кездескен жазбаны сақтап,
# кейінгі толық көшірмелерін жоямыз
df = raw.drop_duplicates().copy()

print("Бастапқы жолдар саны:", len(raw))
print("Дубликаттарды жойғаннан кейінгі жолдар саны:", len(df))

print("\nҚалған артық көшірмелер саны:")
print(df.duplicated().sum())
```

Код түсіндірмесі:

- drop_duplicates() — толық қайталанатын жолдардың артық көшірмелерін алып тастайды;
- .copy() — нәтижені әрі қарай өңдеуге арналған бөлек кесте ретінде көшіреді;

Бастапқы raw кестесі өзгермейді. Өзгерістер df кестесінде орындалады. Жолдар жойылғаннан кейін индекстерде бос орындар болуы қалыпты жағдай. Мысалы, 13-тен кейін 15 келуі мүмкін.

Кодтарды жеке-жеке жазып Кестелерді тексеріп, салыстырыңыз!
Жаңа ұяшыққа!

```
print(df.to_string(index=True))
```

Жаңа ұяшыққа! Жазып тексеріңіз

```
df = df.reset_index(drop=True)
print(df.to_string(index=True))
```

10. Қате мәтіндік мәндерді бос мәнге ауыстыру
Кестедегі unknown және error мәндерін тауып, оларды NaN мәніне ауыстырамыз. Жаңа ұяшыққа жазыңыз:

```
# Қате мәтіндік мәндерді NaN мәніне ауыстырамыз
df = df.replace(["unknown", "error"], float("nan"))

print("Мәтіндік қателер бос мәнге ауыстырылған кесте:")
print(df.to_string(index=True))
```

Бос ұяшықтардың санын көру үшін жаңа ұяшыққа жазыңыз:

```
print("Әр бағандағы бос мәндер саны:")
print(df.isna().sum())
```

isna() бос ұяшықтарды анықтайды, ал sum() олардың санын әр баған бойынша есептейді.
11. Бос мәндерді медианамен толтыру

Жаңа ұяшыққа жазыңыз:

```
# Мәндерді сандық типке келтіреміз
df = df.apply(pd.to_numeric)

# Әр бағанның медианасын есептейміз
medians = df.median()

print("Бағандардың медианалары:")
print(medians)

# Бос ұяшықтарды өз бағанының медианасымен толтырамыз
df = df.fillna(medians)

print("\nБос мәндер толтырылған кесте:")
print(df.to_string(index=True))
```

Түсіндірме
df.apply(pd.to_numeric) – Кестедегі мәндерді сандық типке келтіреді
df.median() – Әр бағанның медианасын жеке есептейді
df.fillna(medians) – Бос ұяшықтарды тиісті бағанның медианасымен толтырады