# Amaliy ish № 6

# Apache Spark bilan ishlash

**Ishning maqsadi:**

Katta ma'lumotlarni tahlil qilish uchun Python (PySpark) orqali Apache Spark-dan foydalanishni o'rganish.

## Nazariy qism:

1. Apache Spark nima?

   Apache Spark - bu taqsimlangan katta ma'lumotlarni qayta ishlash uchun framework. U qattiq disk o'rniga operativ xotiras (RAM) ishlatgani uchun Hadoop'dan tezroq.

   Spark qo'llab-quvvatlaydi:

   - Ommaviy ma'lumotlarni qayta ishlash.
   - Oqimli ma'lumotlarni qayta ishlash (streaming).
   - Mashinani o'qitish (MLlib).
   - Grafik hisoblash (GraphX).

2. PySpark nima?

   PySpark - bu Python uchun Spark interfeysi. Bu sizga tanish Python sintaksisi orqali Spark kuchidan foydalanish imkonini beradi.

   PySpark ning asosiy komponentlari:

   - SparkSession: Spark-ga kirish nuqtasi.
   - DataFrame: satrlar va ustunlar shaklida tashkil etilgan ma'lumotlarning taqsimlangan to'plami (Pandas DataFrame-ga o'xshash, lekin katta ma'lumotlar uchun).

3. Spark nima uchun kerak?

   Spark bitta kompyuter uchun juda katta bo'lgan ma'lumotlarni qayta ishlash uchun ishlatiladi. Masalan:

   - Veb-server jurnallari.
   - Ijtimoiy tarmoqlar.
   - IoT sensorlaridan olingan ma'lumotlar.

## Amaliy qism:

1. PySpark-ni o'rnatish:

   Python (3.8+ versiyasi) o'rnatilganligiga ishonch hosil qiling. Agar PySpark o'rnatilmagan bo'lsa, buyruqni bajaring:

   ```
   pip install pyspark
   ```

2. Ma'lumotlarni tayyorlash:

   Oldingi ishlarda yuklab olgan xuddi shu Titanic (`train.csv`) ma'lumotlar to'plamidan foydalaniladi.

3. Ma'lumotlar bilan ishlash:

   Yangi Python faylini yarating (masalan, `amaliy_5.py`) va quyidagi amallarni bajaring:

```
from pyspark.sql import SparkSession
from pyspark.sql.functions import avg

# SparkSession yaratish
spark = SparkSession.builder \
    .appName("TitanicAnalysis") \
    .getOrCreate()

# Ma'lumotlar yuklash
data = spark.read.csv('train.csv', header=True, inferSchema=True)

# 1. Birinchi 5 qatorni chop eting
print ("Birinchi 5 qator ma'lumotlar:")
data.show(5)

# 2. Kabin sinfi bo'yicha o'rtacha yoshni hisoblash
avg_age_by_class =
data.groupBy('Pclass').agg(avg('Age').alias('Average_Age'))
print("\Samolyot sinfi bo'yicha yo'lovchilarning n o'rtacha
yoshi:")
avg_age_by_class.show()

# 3. Omon qolganlar sonini hisoblash
survived_count = data.groupBy('Survived').count()
print ("\Omon qolganlar va o'lganlar n soni:")
```

```
survived_count.show()

# SparkSessionni tugatish
spark.stop()
```

4. Dasturni ishga tushirish:

   Faylni saqlang va uni terminal orqali ishga tushiring:

   ```
   python practical_5.py
   ```

   Natijada ko'rasiz:

   - Ma'lumotlarning dastlabki 5 qatori.
   - Kabin sinfi bo'yicha yo'lovchilarning o'rtacha yoshi.
   - Omon qolganlar va o'lganlar soni.

**Qo'shimcha materiallar:**

- Rasmiy PySpark hujjatlari: PySpark Documentation .
- Spark o'rnatish bo'yicha qo'llanma: Spark Installation Guide .
- PySpark-dan foydalanishga misollar: PySpark Tutorials .

## Amaliy vazifa:

*Vazifa: № 5*

Titanic (`train.csv`) ma'lumotlar to'plami va PySpark dan foydalanib, quyidagi amallarni bajaring:

1. Har bir kabina sinfi uchun (`Pclass`) maksimal chipta narxini toping (`Fare`)
2. Omon qolganlar orasida erkaklar va ayollar sonini hisoblang.
3. Har bir jins kombinatsiyasi (`Sex`) va kabina sinfi (`Pclass`) uchun o'rtacha chipta narxini ko'rsatadigan jadval tuzing