Департамент образования и науки города Москвы
Государственное автономное образовательное учреждение
высшего образования города Москвы
«Московский городской педагогический университет»
Институт цифрового образования
Департамент информатики управления и технологий

Агафонов Антон Александрович БД-241м

# Практическая работа 1. Введение в большие данные и их хранение. Инструменты обработки больших данных (Hadoop)

Направление подготовки/специальность
38.04.05 - Бизнес-информатика
Бизнес-аналитика и большие данные
(очная форма обучения)
Вариант 1

Москва
2025

Цель

Изучить основные операции и функциональные возможности системы, что позволит понять принципы работы с данными и распределенными вычислениями.

## Основная часть

Запускаем hadoop

**Start-dfs.sh**

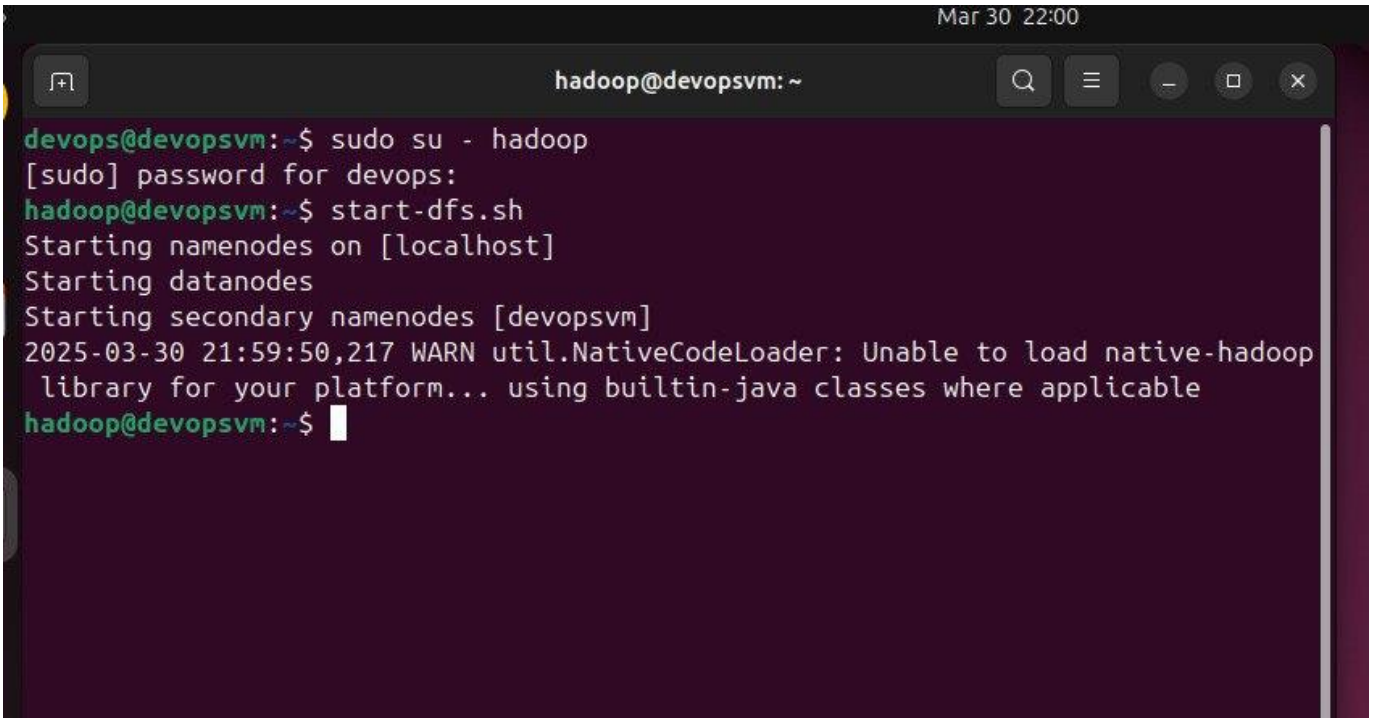


**Рис.1** Запускаем файловую систему

**Start-yarn.sh**

Проверяем запущенные службы командой

**jps**

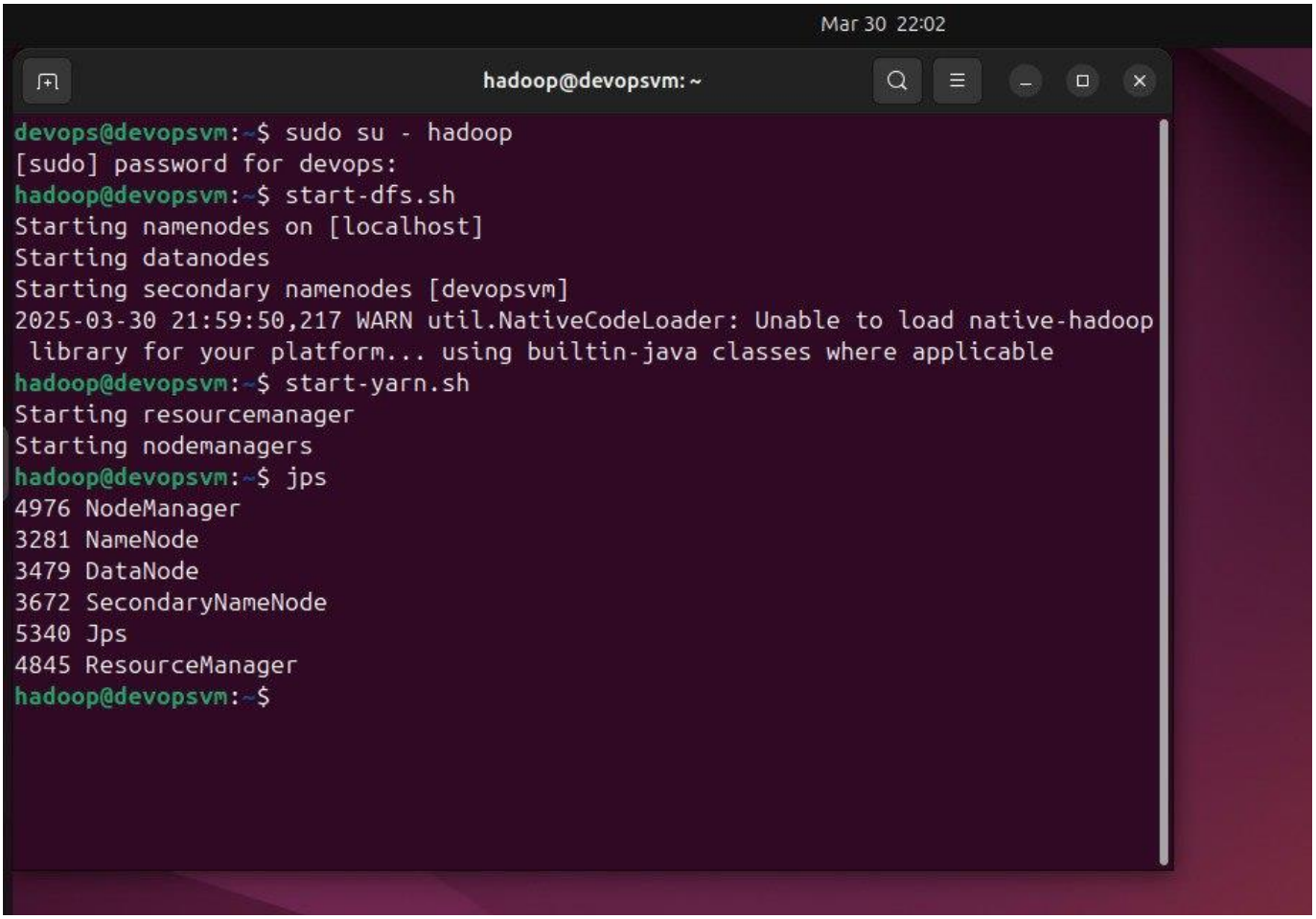


Рис.2 Проверка запущенных служб

Проверяем доступность запущенных систем

Переходим по ссылке для проверки запущен ли dfs по ссылке

**Localhost:9870/dfshealth.html#tab-overview**

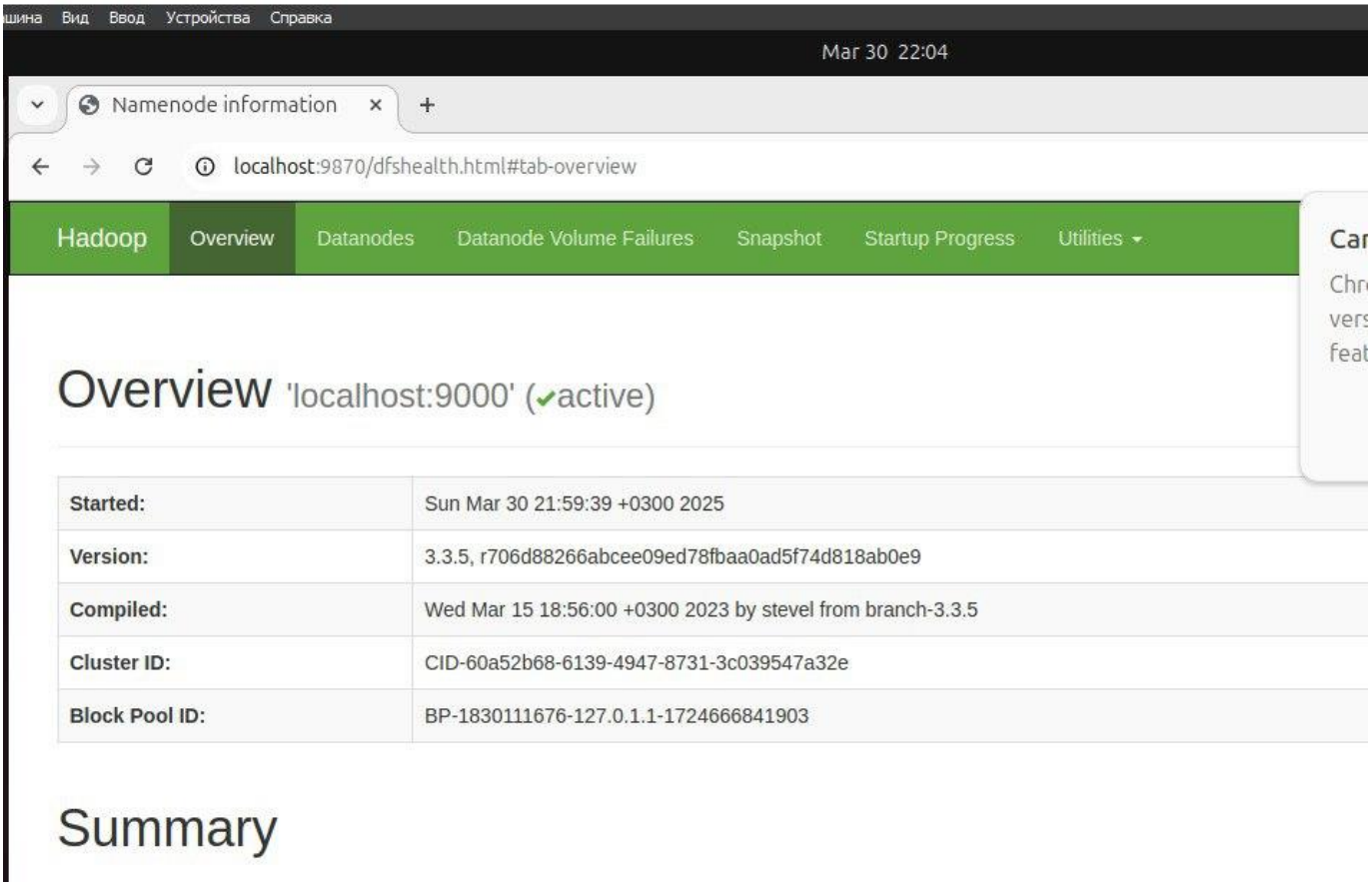


Рис.3 проверка доступности систем

Проверяем запущен ли yarn по ссылке

**Localhost:8088/cluster**

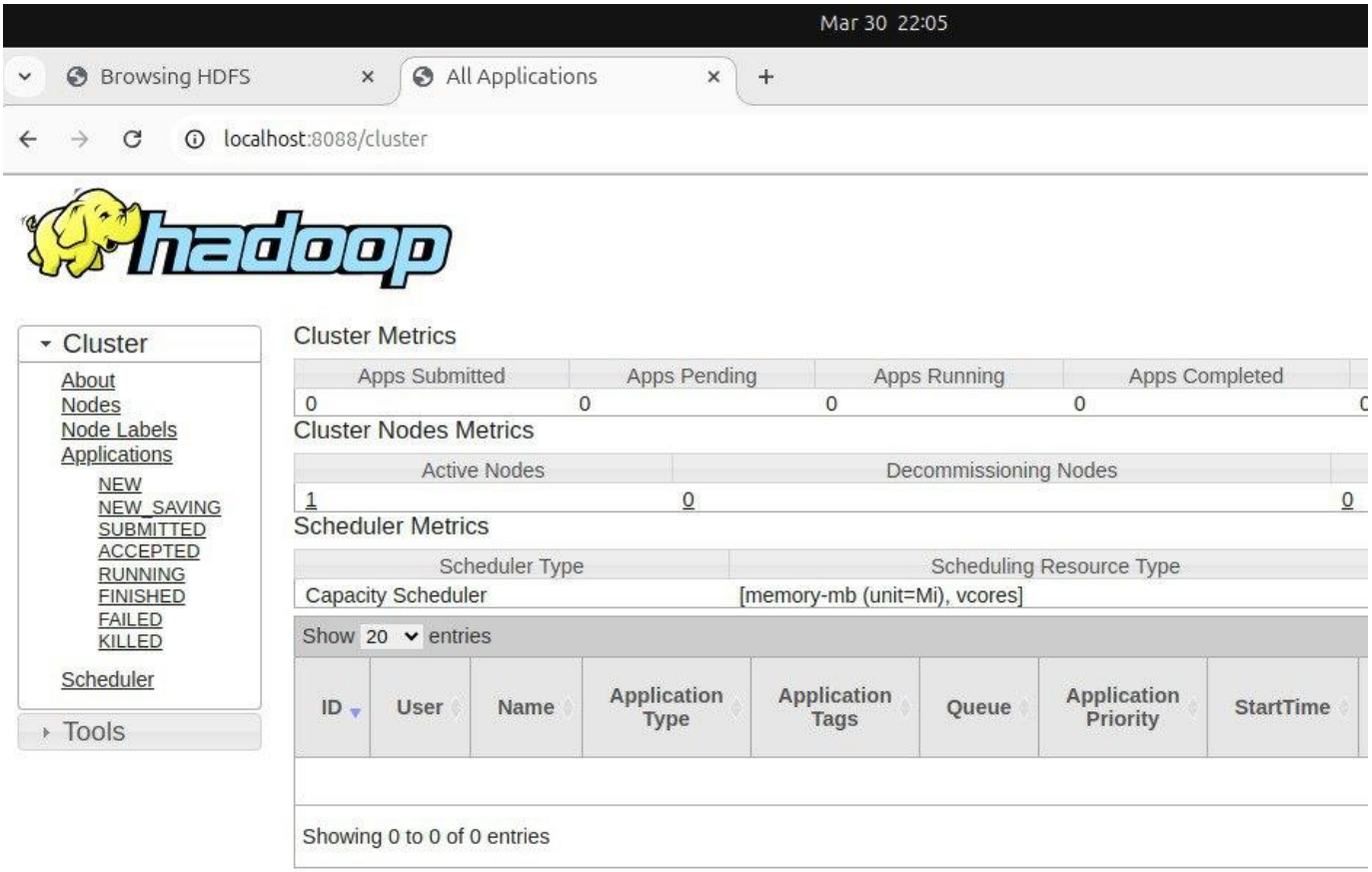


**Рис.4 Проверка работы yarn**

Создаем пользователя и каталог командой

**Hdfs dfs -mkdir -p /agafonov01/Hadoop/input**

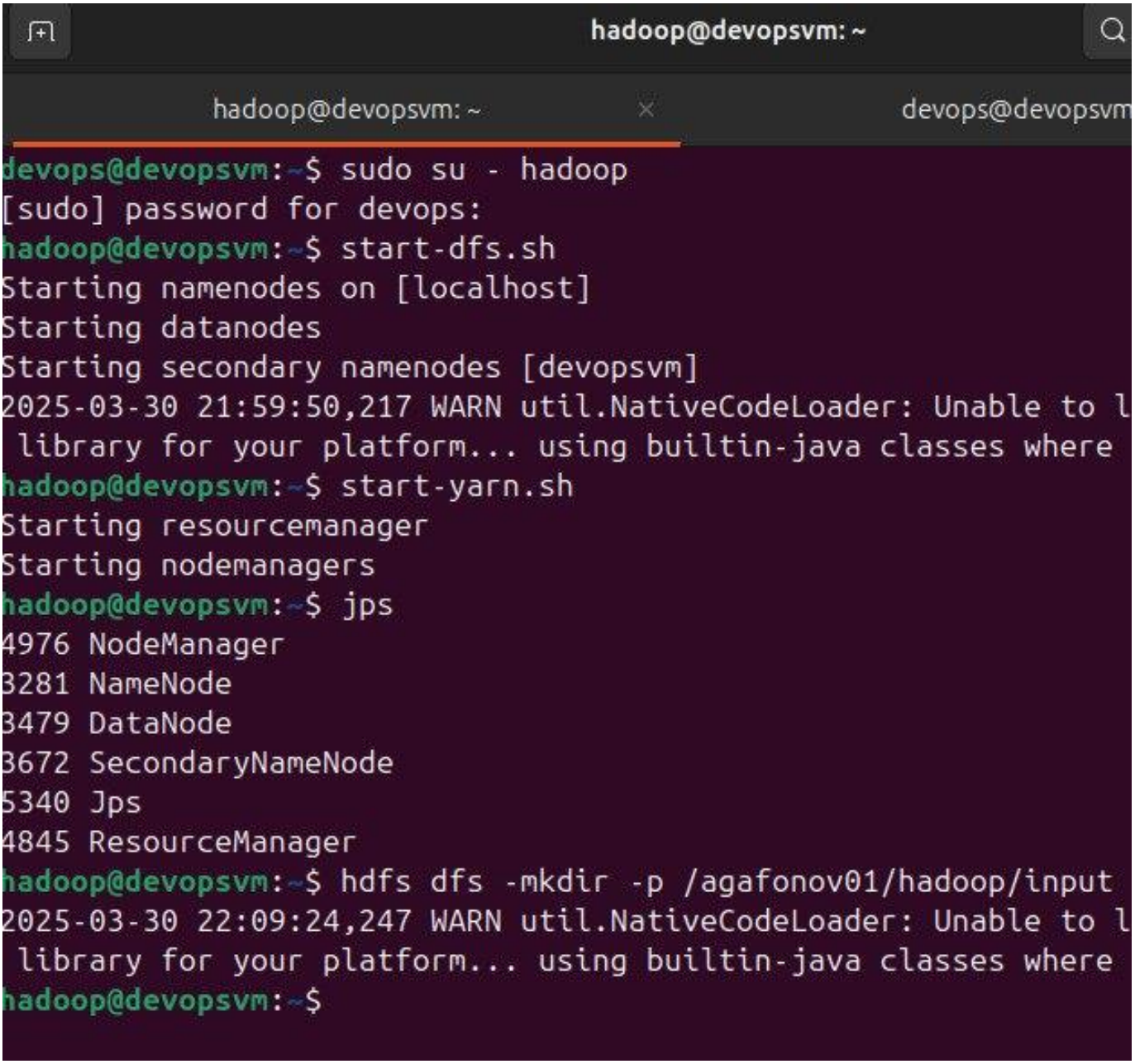


**Рис.5 Создание пользователя**

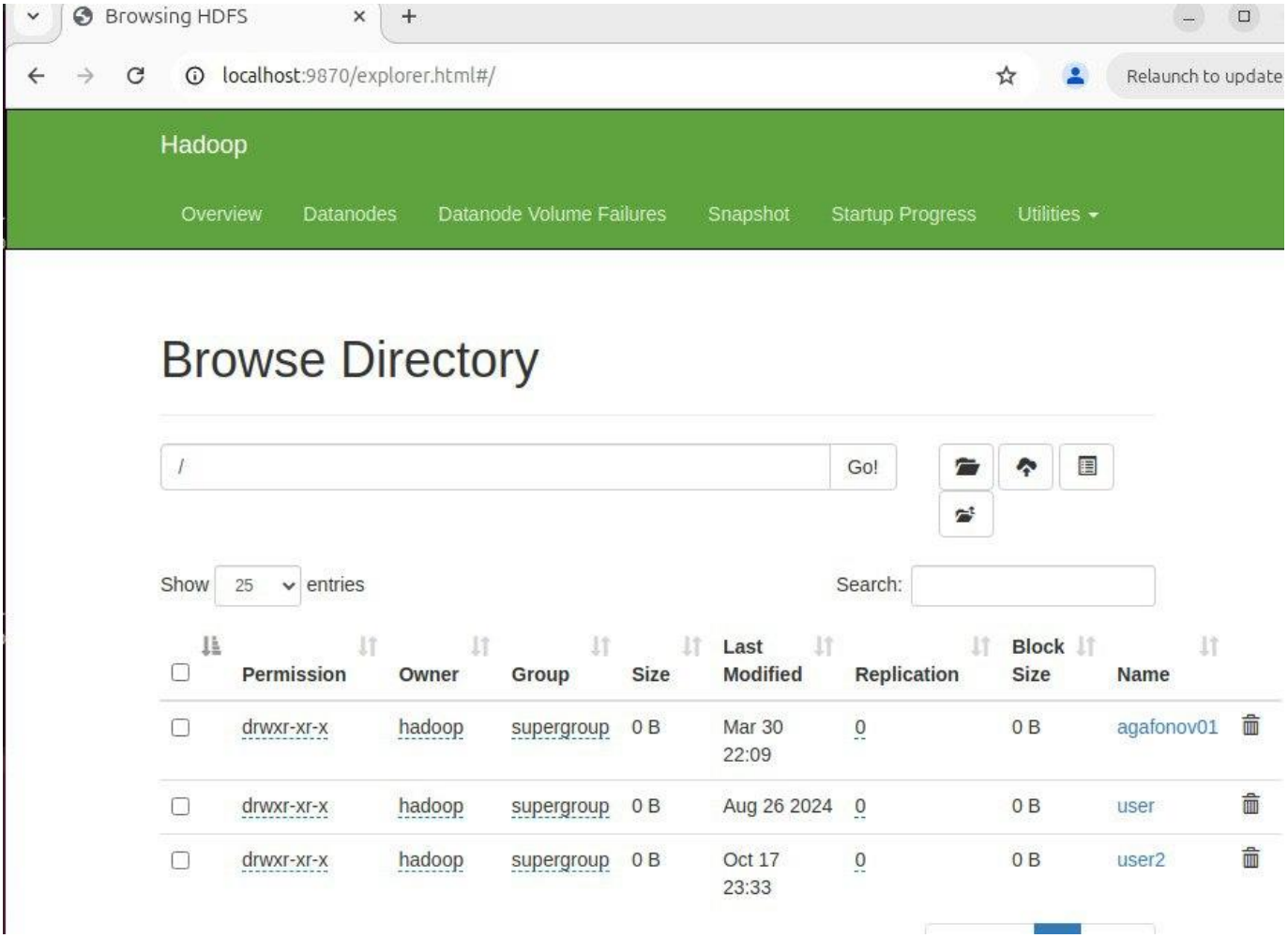


**Рис.6 Проверка созданного пользователя**

**Скачиваем данные**

**wget https://raw.githubusercontent.com/BosenkoTM/Distributed_systems/main/practice/2024/l w_01/GDP.csv**

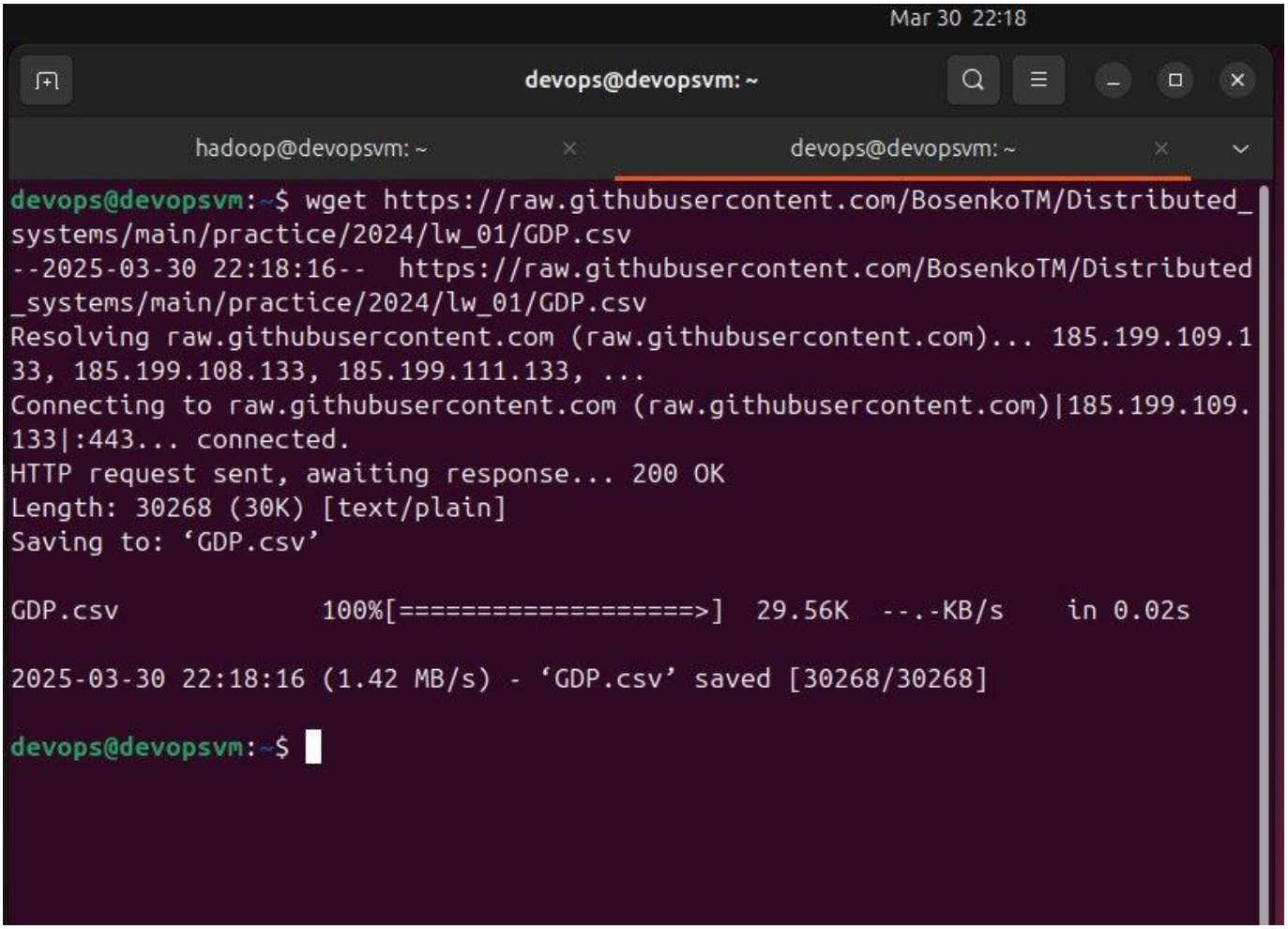


**Рис.7 Скачиваем данные**

Создаем каталог для данных

**Hdfs dfs -mkdir -p /agafonov01/Hadoop/input/economic_data**

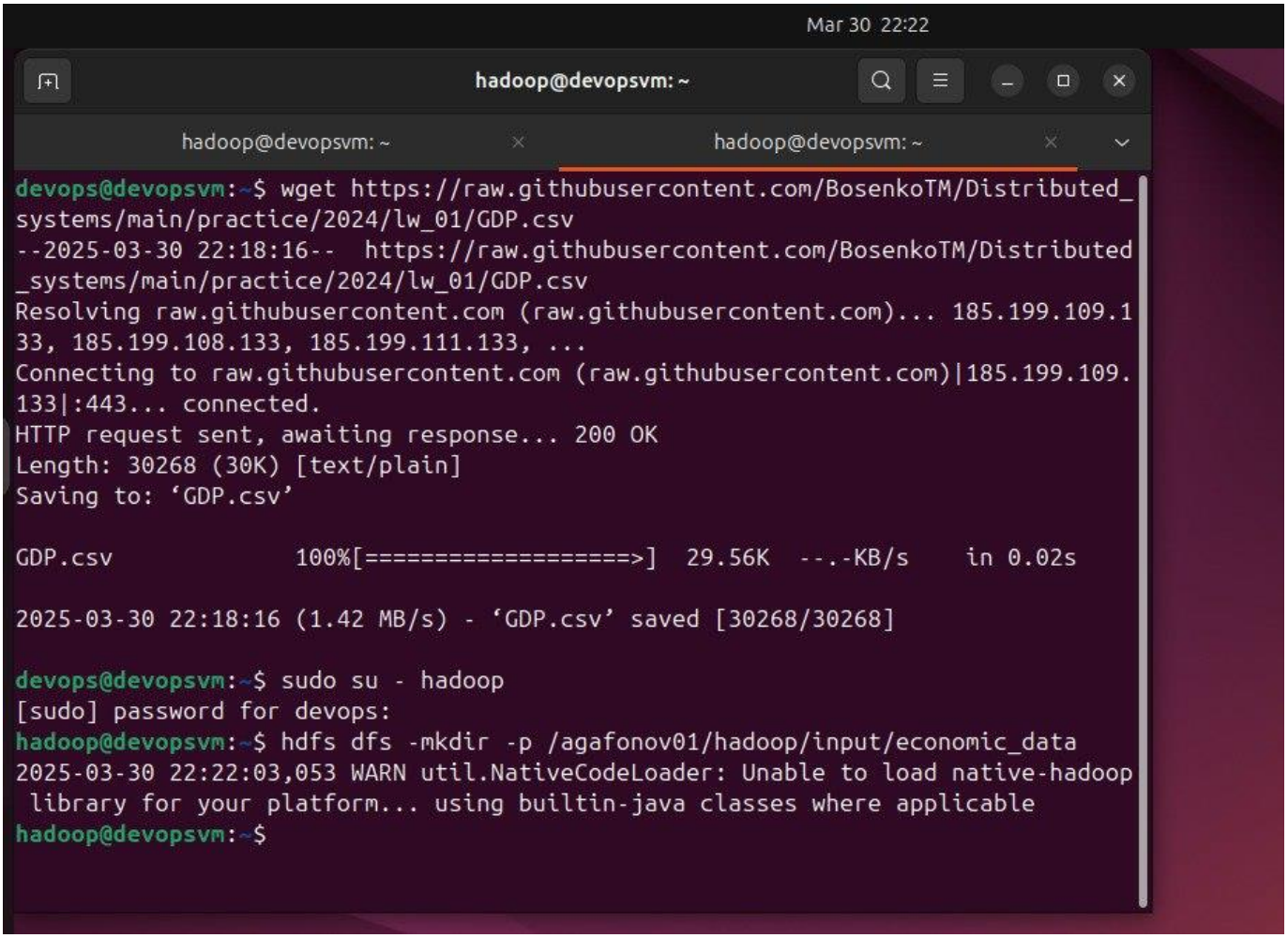


**Рис.8 Создание каталога economic_data**

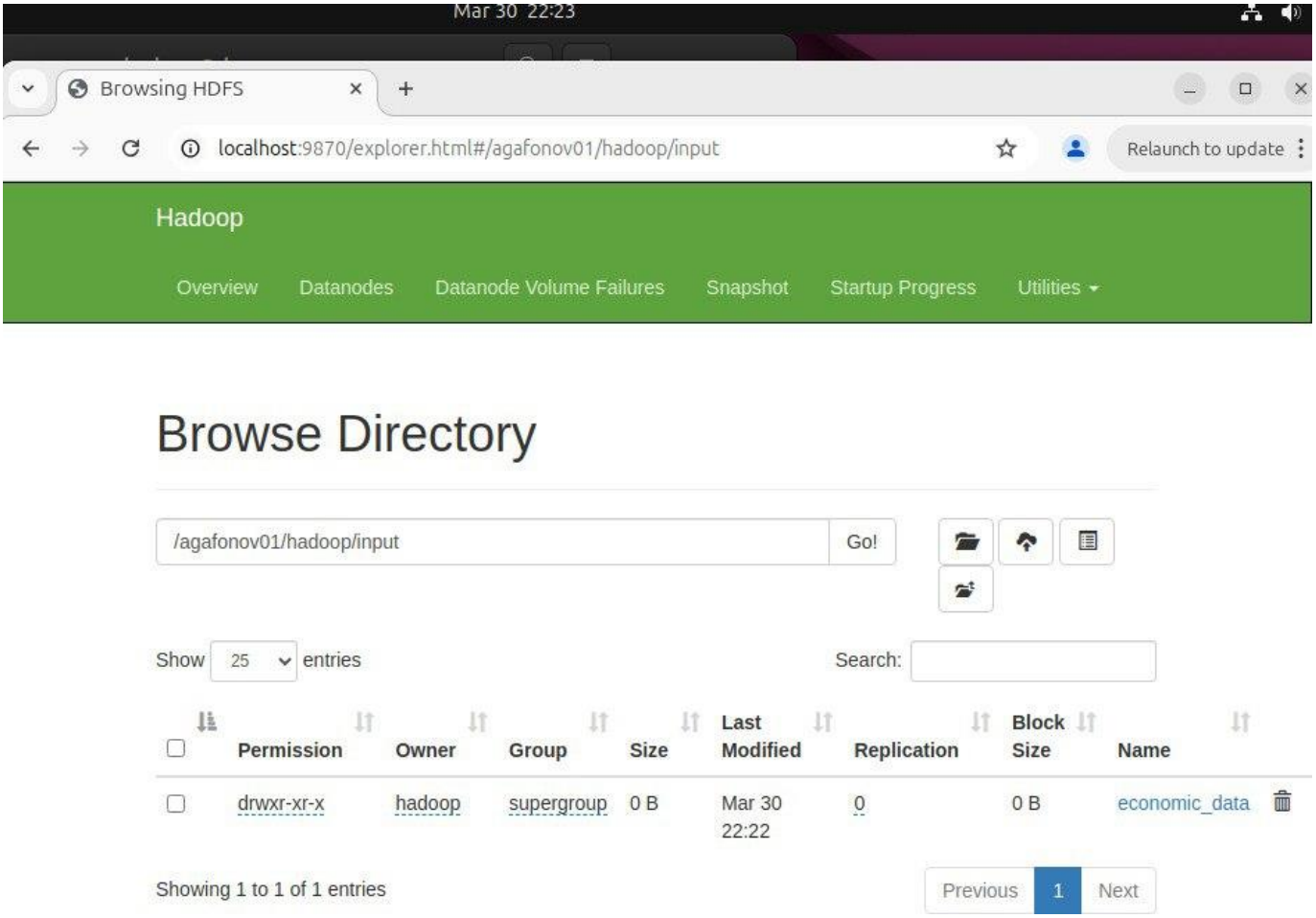


**Рис.9 Проверка созданного каталога economic_data**

**Переносим данные в каталог**

**Hdfs dfs -put GDP.csv /agafonov01/Hadoop/input/economic_data**

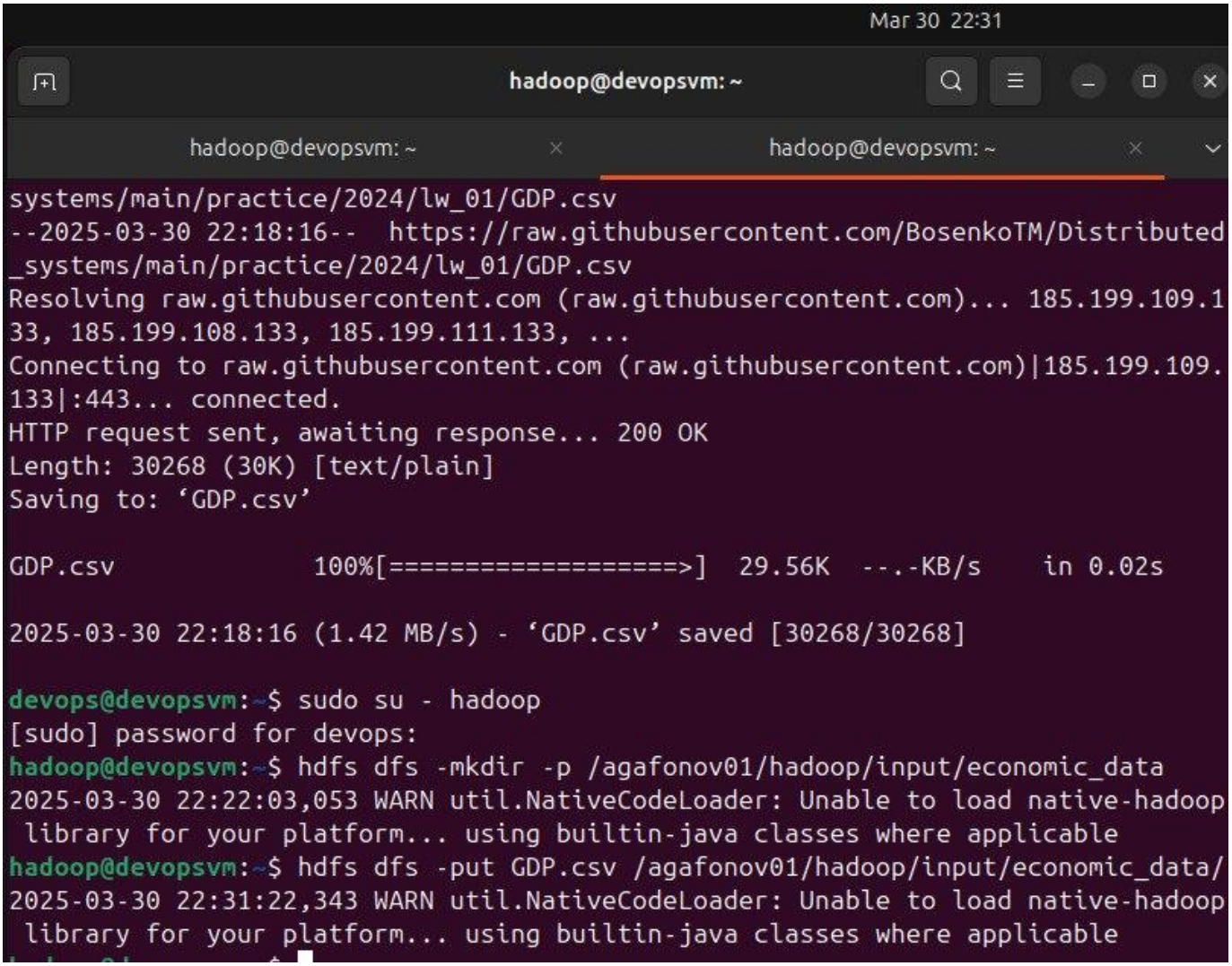


**Рис.10 Перенос данных в каталог**

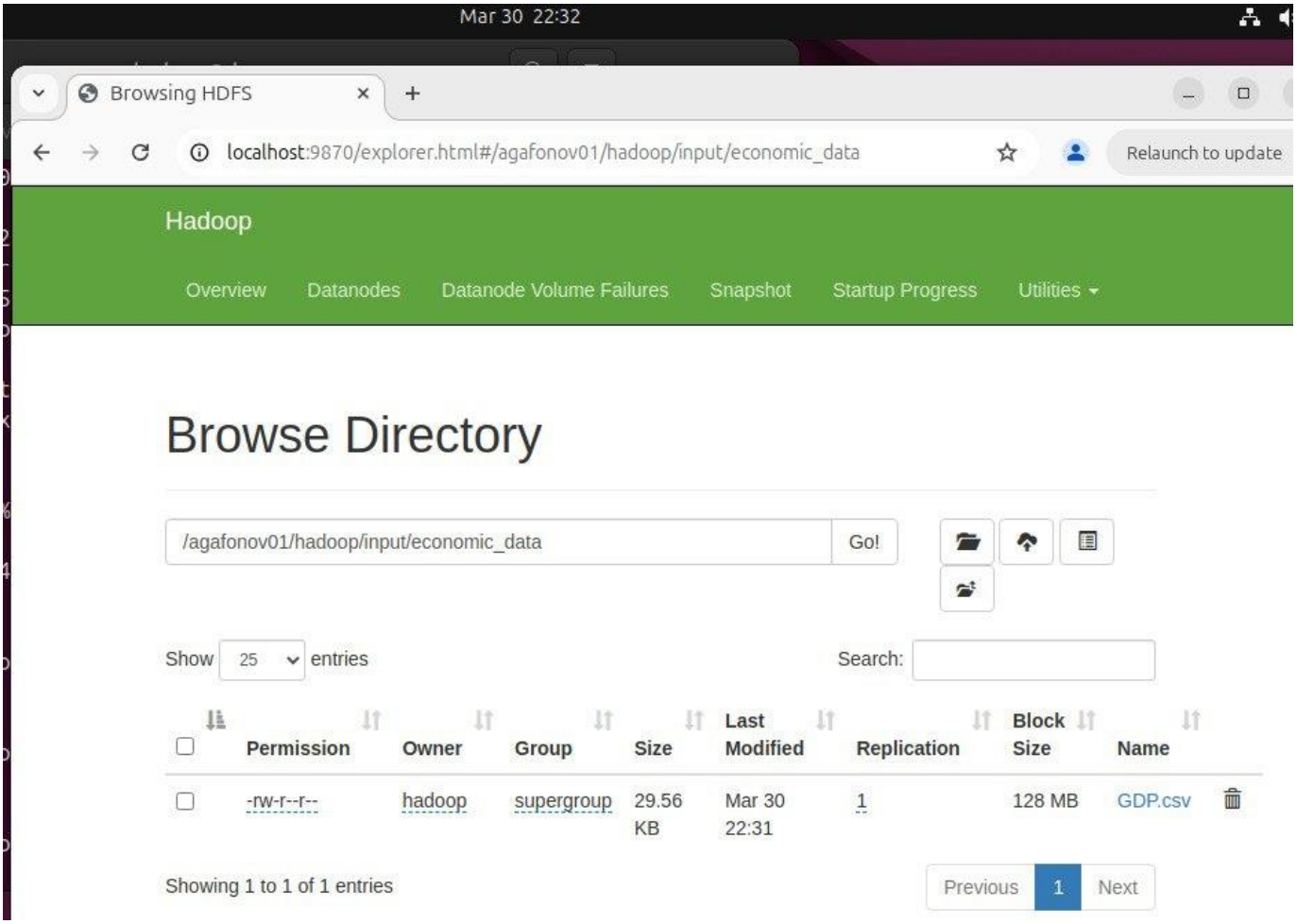


**Рис.11 Проверка перенесенных данных**

Задаем права доступа

**hdfs dfs -chmod 777 /agafonov01/hadoop/input/economic_data**

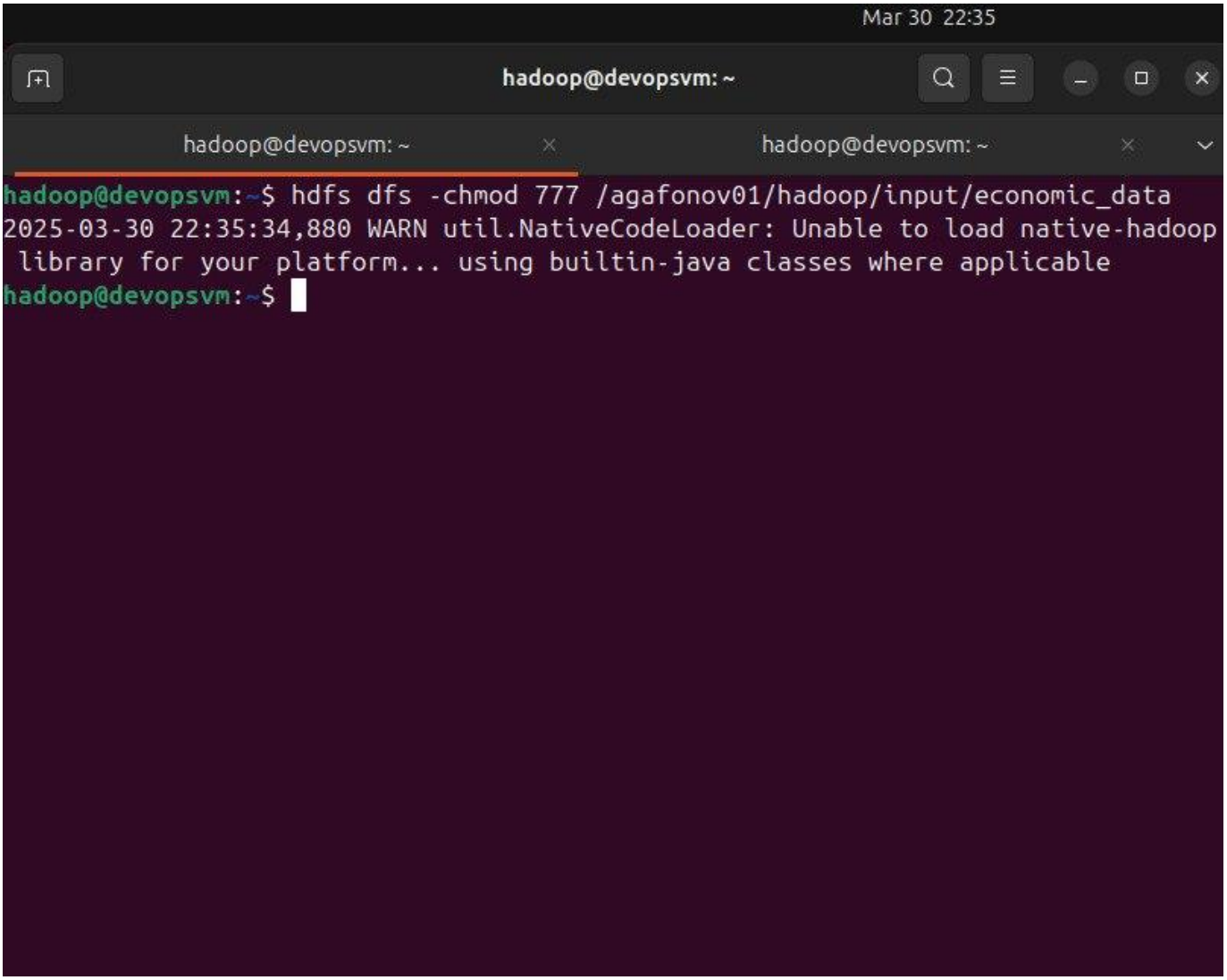


**Рис.12** Устанавливаем права доступа

**Обрабатываем данные при помощи Spark**

**Spark-shell**

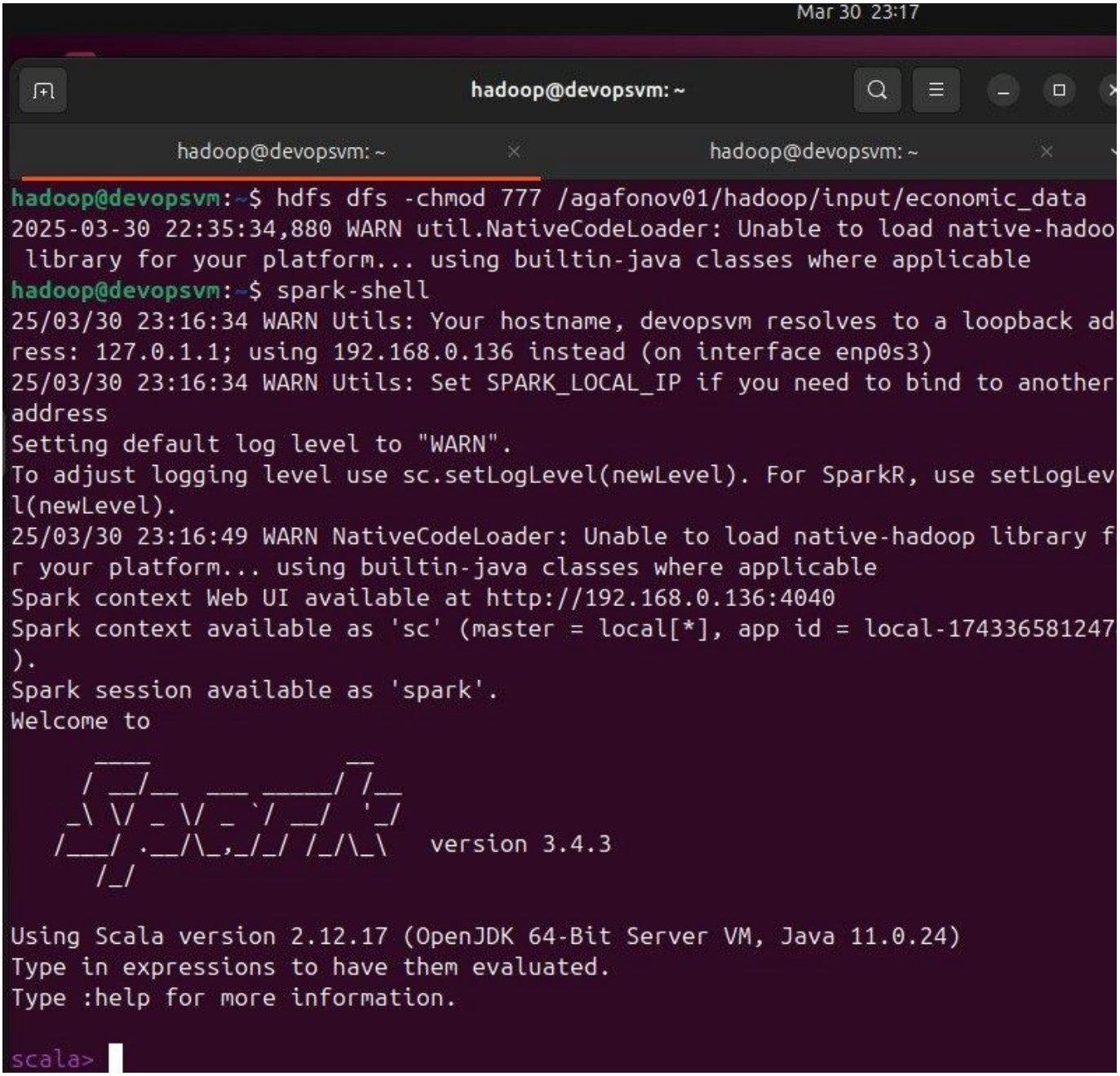


**Рис.13** Запуск spark

Загружаем данных из hdfs

**val data = spark.read.option("header", "true").csv("file:///home/hadoop/GDP.csv")**

```
scala> val data = spark.read.option("header","true").csv("file:///home/hadoop/GD
P.csv")
data: org.apache.spark.sql.DataFrame = [Country: string, Year: string ... 21 mor
e fields]

scala>
```

**Рис.14** Загрузка данных

Проверка полученной схемы данных

**data.printSchema()**

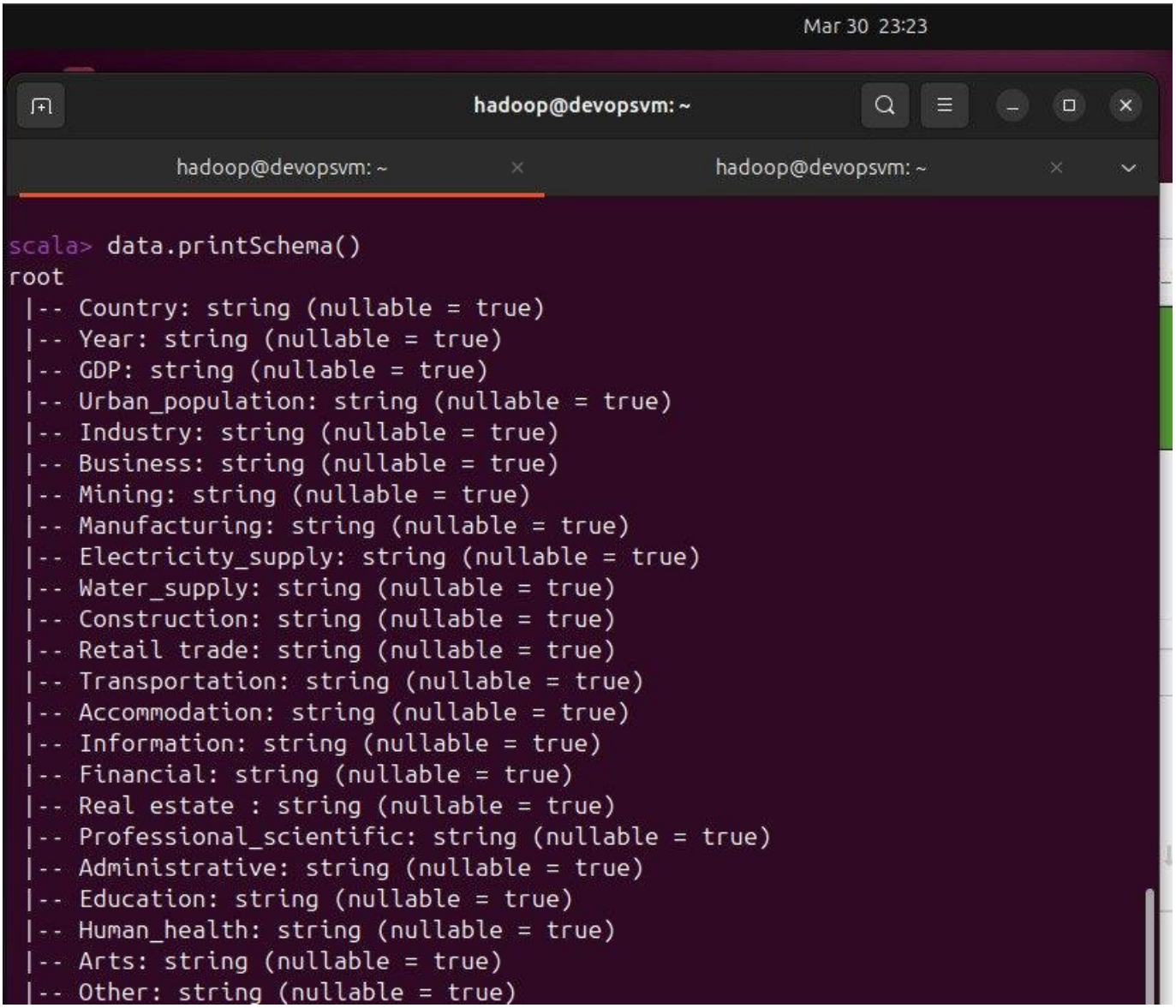


**Рис.15** Выводим схему

Вычисление среднего значения GDP

**val result = data.selectExpr("avg(GDP) as avg_GDP")**

```
scala> val result=data.selectExpr("avg(GDP) as avg_GDP")
result: org.apache.spark.sql.DataFrame = [avg_GDP: double]

scala>
```

**Рис.16** Вычисление среднего значения GDP

Сохраняем результата в CSV файл

**result.write.option("header", "true").csv("/home/hadoop/output/avg_GDP.csv")**

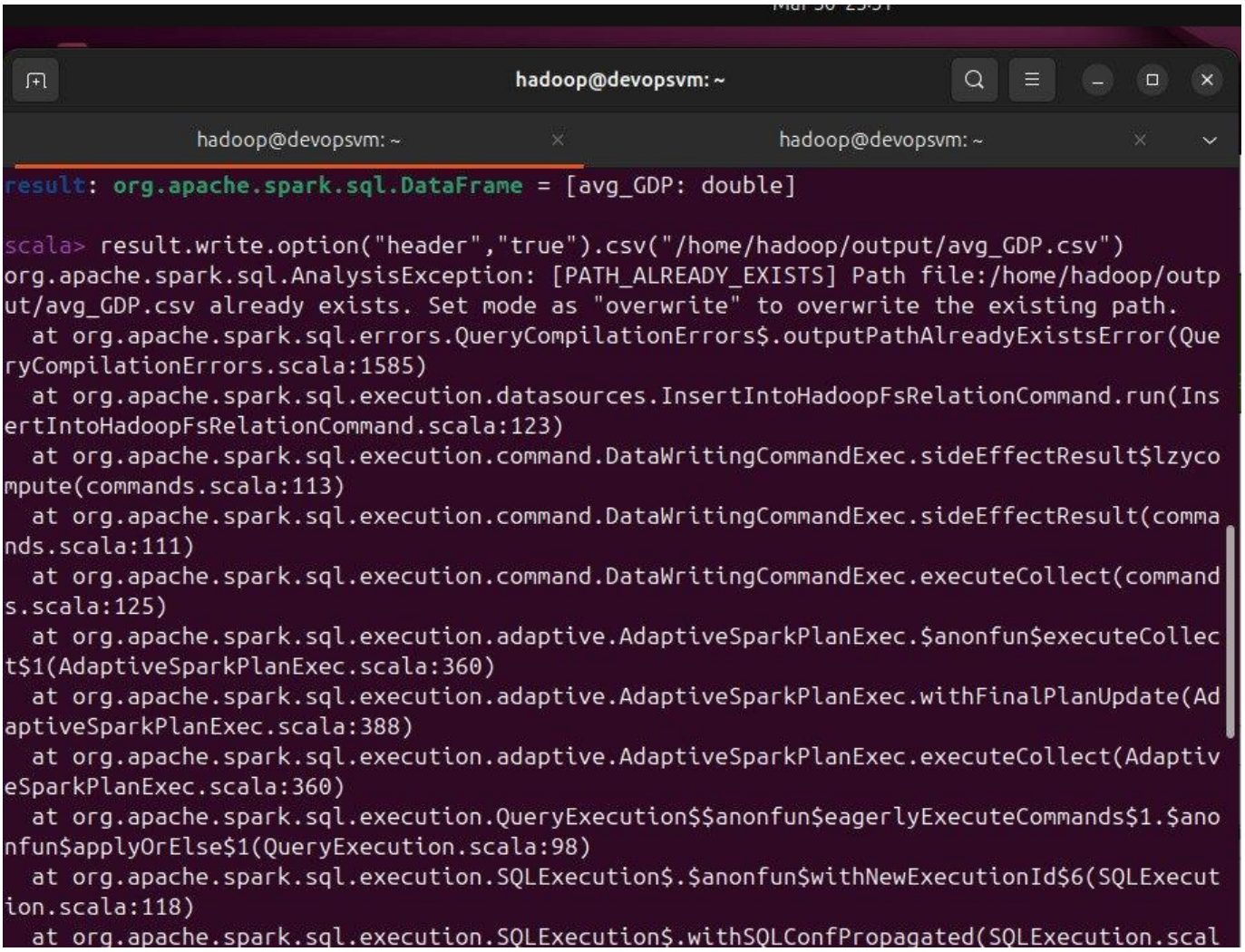


**Рис.17** сохранение результатов

Переходим в директорию с результатами

**cd /home/hadoop/output/avg_GDP.csv**

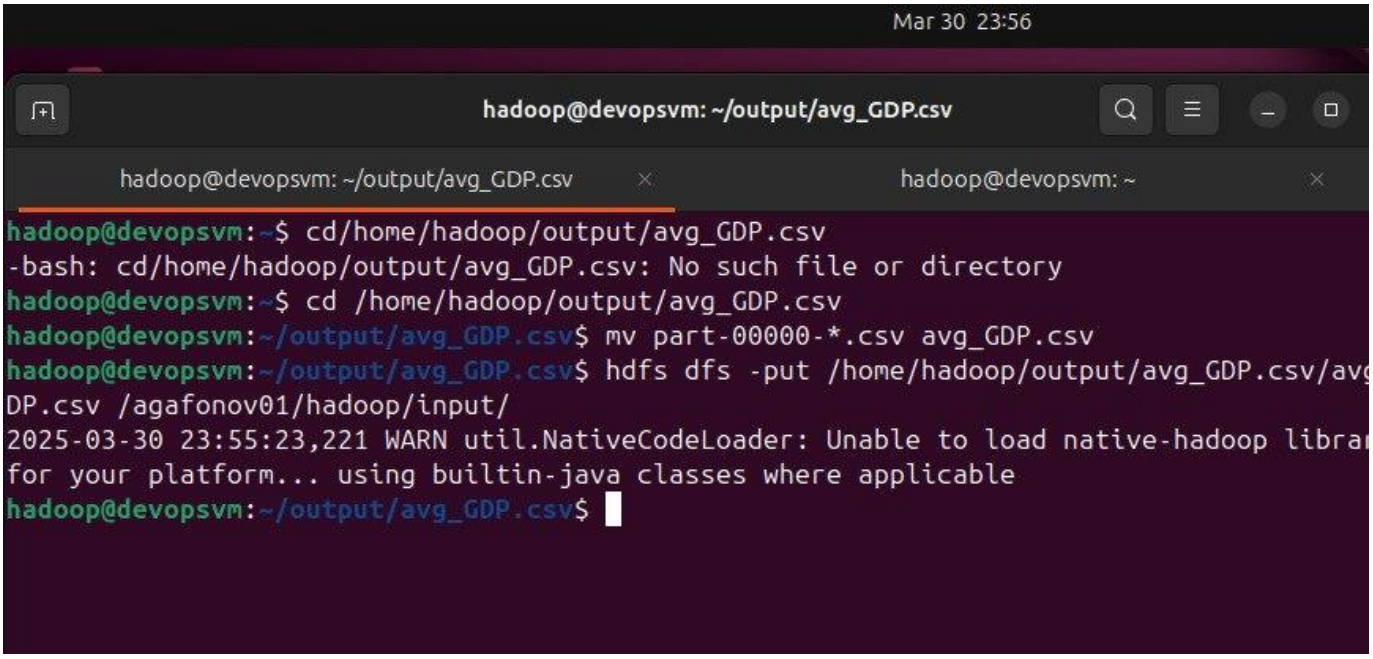


**Рис.18 Проверяем полученный файл**

Загружаем полученный файл в HDFS

**hdfs dfs -put /home/hadoop/output/avg_GDP.csv /agafonov01/hadoop/input/**

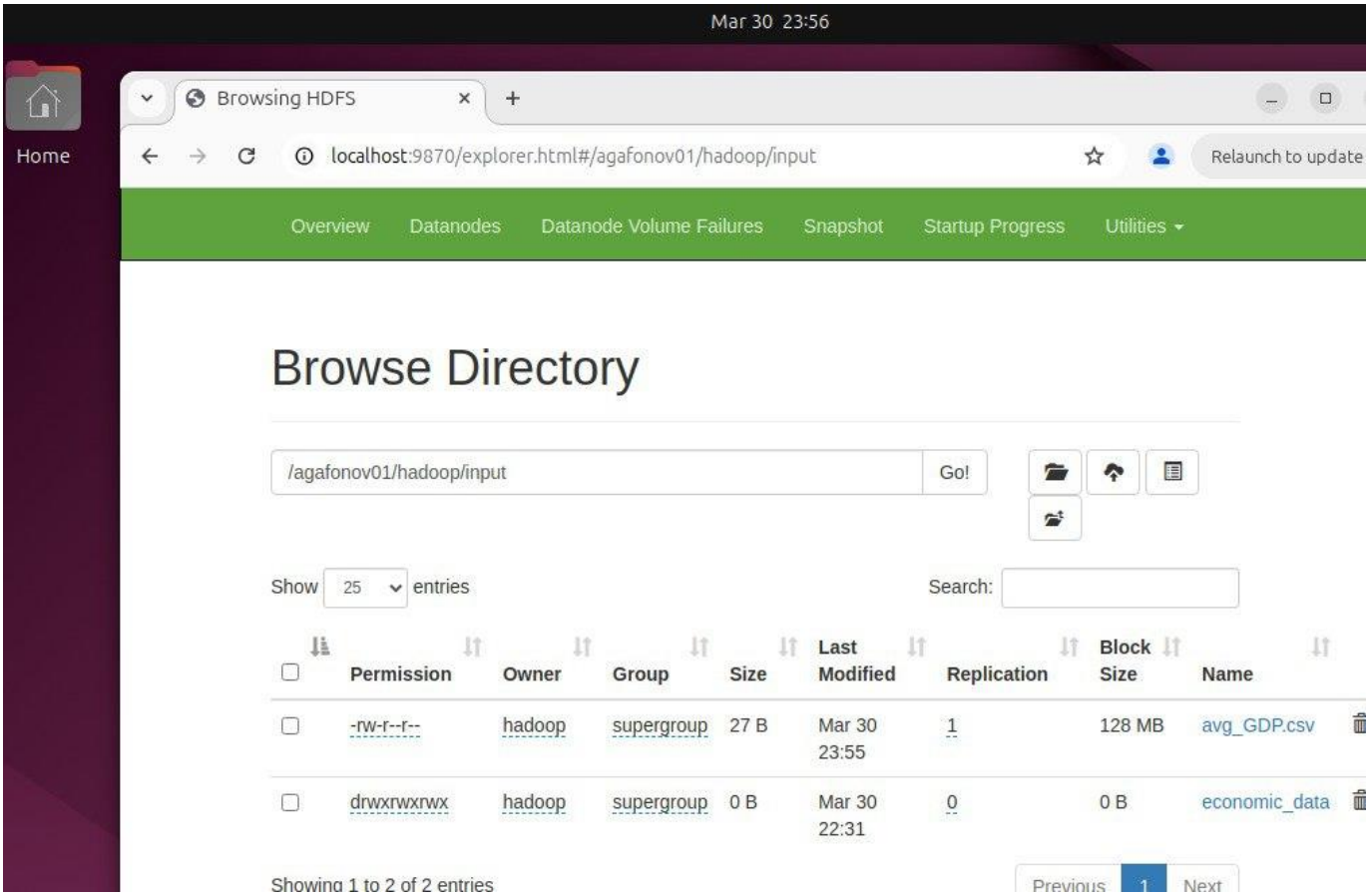


**Рис.19 Проверка загруженного файла в hdfs**

Проверка загрузки

**hdfs dfs -ls /user01/hadoop/input/**

```
hadoop@devopsvm:~/output/avg_GDP.csv$ hdfs dfs -ls /agafonov01/hadoop/input
2025-03-30 23:57:42,223 WARN util.NativeCodeLoader: Unable to load native-hadoop library
for your platform... using builtin-java classes where applicable
Found 2 items
-rw-r--r--   1 hadoop supergroup         27 2025-03-30 23:55 /agafonov01/hadoop/input/avg
_GDP.csv
drwxrwxrwx   - hadoop supergroup          0 2025-03-30 22:31 /agafonov01/hadoop/input/eco
nomic_data
hadoop@devopsvm:~/output/avg_GDP.csv$
```

**Рис.20 Проверка загрузки**

## Задание для самостоятельной работы

Подключиться к HDFS и убедиться, что файл доступен по пути hdfs://localhost:9000/agafonov01/hadoop/economic_data/GDP.csv

Запускаем jupyterlab и загружаем файлы из hdfs

**File_path= “hdfs://localhost:9000/agafonov01/Hadoop/input/economic_data/GDP.csv”**

**df = spark.read.csv(file_path, header=True, inferShema =True )**

**df.show(5)**

[38]:
```
# Чтение данных из HDFS
file_path = "hdfs://localhost:9000/agafonov01/hadoop/input/economic_data/GDP.csv"
df = spark.read.csv(file_path, header=True, inferSchema=True)

# Просмотр первых строк данных
df.show(5)
```

```
+-------+----+-----+----------------+--------+--------+------+-------------+------------------+------------+------
------+------------+--------------+-------------+-----------+---------+------------+----------------------+------
--------+---------+------------+----+-----+
|Country|Year|  GDP|Urban_population|Industry|Business|Mining|Manufacturing|Electricity_supply|Water_supply|Constr
uction|Retail trade|Transportation|Accommodation|Information|Financial|Real estate |Professional_scientific|Admini
strative|Education|Human_health|Arts|Other|
+-------+----+-----+----------------+--------+--------+------+-------------+------------------+------------+------
------+------------+--------------+-------------+-----------+---------+------------+----------------------+------
--------+---------+------------+----+-----+
|Austria|2010|35390|           57.4|    24.0|    25.2|  18.3|         24.4|              23.6|        12.2|
9.9|        27.5|          7.3|          9.9|       21.2|     30.3|        27.0|                  34.0|
22.5|     27.8|        12.0|34.0| 32.0|
|Austria|2015|36140|           57.72|    21.8|    23.4|  13.7|         22.7|              17.6|         9.3|
8.2|        23.3|         11.6|          6.4|       22.4|     30.3|        28.0|                  31.3|
20.0|     24.2|        12.9|26.2| 28.3|
|Austria|2016|36390|           57.91|    20.8|    22.3|  14.4|         21.9|              13.2|         8.2|
8.3|        23.3|         14.5|          5.9|       20.9|     27.1|        28.7|                  30.4|
17.8|     24.3|        14.5|20.8| 27.8|
|Austria|2017|36980|           58.09|    20.7|    22.3|  10.9|         21.7|              13.0|         8.4|
8.3|        23.2|         12.4|          5.7|       20.6|     28.4|        29.0|                  29.4|
17.4|     23.7|        15.0|19.1| 26.9|
|Austria|2018|37690|           58.3|    20.4|    22.0|   7.9|         21.4|              14.4|         8.1|
8.3|        23.2|         11.7|          5.4|       20.7|     28.2|
17.1|     23.6|        15.3|18.3| 26.4|
+-------+----+-----+----------------+--------+--------+------+-------------+------------------+------------+------
------+------------+--------------+-------------+-----------+---------+------------+----------------------+------
--------+---------+------------+----+-----+
```

**Индивидуальное задание**

**Вариант 1**

**https://raw.githubusercontent.com/BosenkoTM/Distributed_systems/main/practice/2024/l w_01/AAPL.csv**

Для начала создаем каталог в hdfs

**Hdfs dfs -mkdir -p /agafonov01/hadoop/input/vol**


**Рис.21 Создание каталога surgutneftgaz в hdfs**

Переносим файл из пользователя devops в пользователя Hadoop

**sudo mv /home/devops/Desktop/AAPL.csv /home/hadoop**

Переносим данные в каталог

**Hdfs dfs -put AAPL.csv /agafonov01/Hadoop/input/vol**

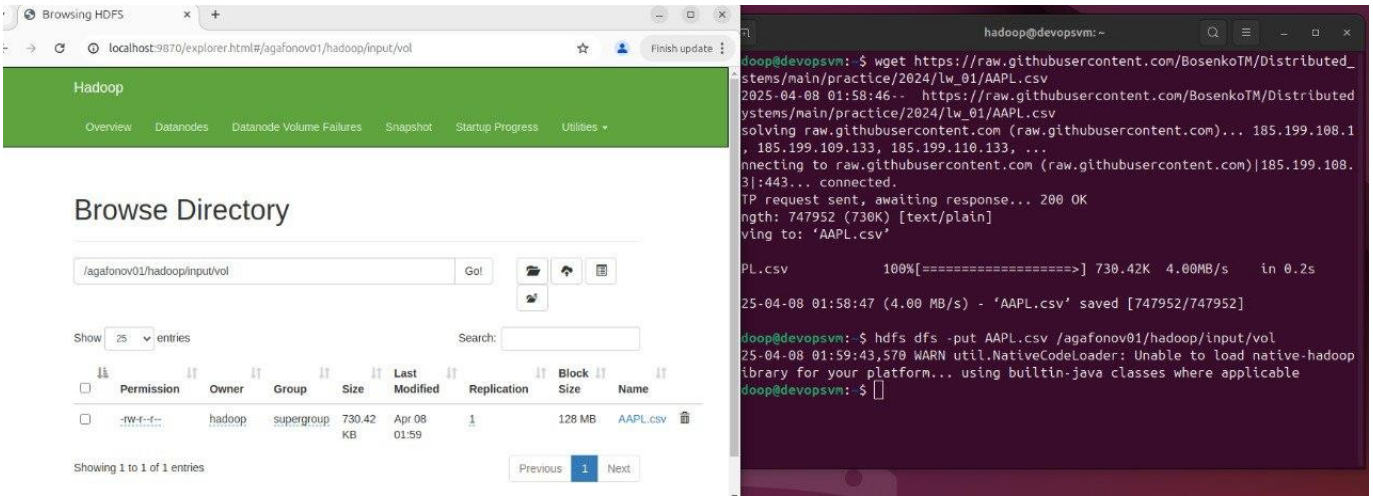


**Рис.22 Переносим данные об акциях в hdfs**

Устанавливаем права доступа

**hdfs dfs -chmod 777 /agafonov01/hadoop/input/vol**

```
hadoop@devopsvm:~$ hdfs dfs -chmod 777 /agafonov01/hadoop/input/vol
2025-04-08 02:02:08,618 WARN util.NativeCodeLoader: Unable to load native-hadoo
 library for your platform... using builtin-java classes where applicable
hadoop@devopsvm:~$
```

**Рис.23** Устанавливаем права доступа

Загружаем данные из hdfs

Import pandas as pd

Import matplotlib.pyplot as plt

From pyspart.sql import SparkSession

Spark = SparkSession.builder\

.appName(“Economic Data Analysis”)\

.config(“spark.hadoop.fs.defaultFS”,”hdfs://localhost:9000”)\

.get0Create()

Spark.conf.set(“spark.sql.shuffle.partition”, “50”)

File_path = “hdfs://localhost:9000/agafonov01/Hadoop/input/vol/AAPL.csv

Df= spark.read.csv(file_path, header = True, inferSchema =True)

Df.show(5)

```
# Создание SparkSession
spark = SparkSession.builder \
    .appName("Economic Data Analysis") \
    .config("spark.hadoop.fs.defaultFS", "hdfs://localhost:9000") \
    .config("spark.ui.port", "4050") \
    .getOrCreate()

# Установка количества разделов для shuffle операций
spark.conf.set("spark.sql.shuffle.partitions", "50")
```

```
# Чтение данных из HDFS
file_path = "hdfs://localhost:9000/agafonov01/hadoop/input/vol/AAPL.csv"
df = spark.read.csv(file_path, header=True, inferSchema=True)

# Просмотр первых строк данных
df.show(5)
```

```
pandas_df = df.toPandas()
pandas_df.head()
```

| | Country | Year | GDP | Urban_population | Industry | Business | Mining | Manufacturing | Electricity_supply | Water_supply | ... | Accommodation | Information | Financial | Real estate | Professional_scientific | Administrative | Education | Human_health | Arts | Othe |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | Austria | 2010 | 35390 | 57.40 | 24.0 | 25.2 | 18.3 | 24.4 | 23.6 | 12.2 | ... | 9.9 | 21.2 | 30.3 | 27.0 | 34.0 | 22.5 | 27.8 | 12.0 | 34.0 | 32. |
| 1 | Austria | 2015 | 36140 | 57.72 | 21.8 | 23.4 | 13.7 | 22.7 | 17.6 | 9.3 | ... | 6.4 | 22.4 | 30.3 | 28.0 | 31.3 | 20.0 | 24.2 | 12.9 | 26.2 | 28. |
| 2 | Austria | 2016 | 36390 | 57.91 | 20.8 | 22.3 | 14.4 | 21.9 | 13.2 | 8.2 | ... | 5.9 | 20.9 | 27.1 | 28.7 | 30.4 | 17.8 | 24.3 | 14.5 | 20.8 | 27. |
| 3 | Austria | 2017 | 36980 | 58.09 | 20.7 | 22.3 | 10.9 | 21.7 | 13.0 | 8.4 | ... | 5.7 | 20.6 | 28.4 | 29.0 | 29.4 | 17.4 | 23.7 | 15.0 | 19.1 | 26. |
| 4 | Austria | 2018 | 37690 | 58.30 | 20.4 | 22.0 | 7.9 | 21.4 | 14.4 | 8.1 | ... | 5.4 | 20.7 | 28.2 | 29.2 | 28.3 | 17.1 | 23.6 | 15.3 | 18.3 | 26. |

5 rows × 23 columns

## Заключение

В ходе проделанной лабораторной работы, были изучены основные операции и функциональные возможности системы, что позволило понять принципы работы с данными и распределенными вычислениями, также было выполнено задание по вариантам.