**Xuất dữ liệu các bảng vào HDFS**

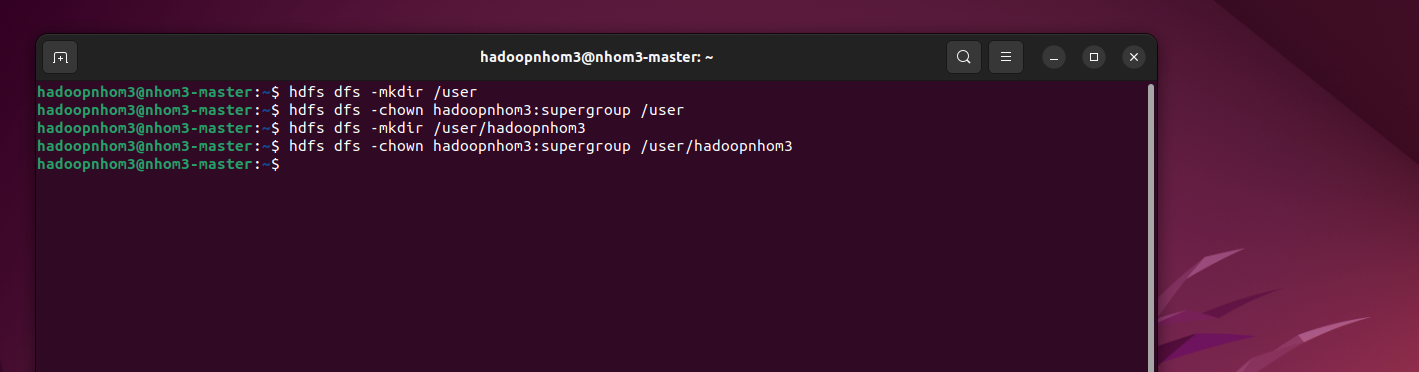
Tạo thư mục chứa dữ liệu

hdfs dfs -mkdir /user

hdfs dfs -chown hadoopnhom3:supergroup /user

hdfs dfs -mkdir /user/hadoopnhom3

hdfs dfs -chown hadoopnhom3:supergroup /user/hadoopnhom3



**Xuất Bản QUANAN**

sqoop import \

--connect jdbc:mysql://localhost:3306/QuanLyQuanAn \

--driver com.mysql.jdbc.Driver \

--username sqoopdb \

-P \

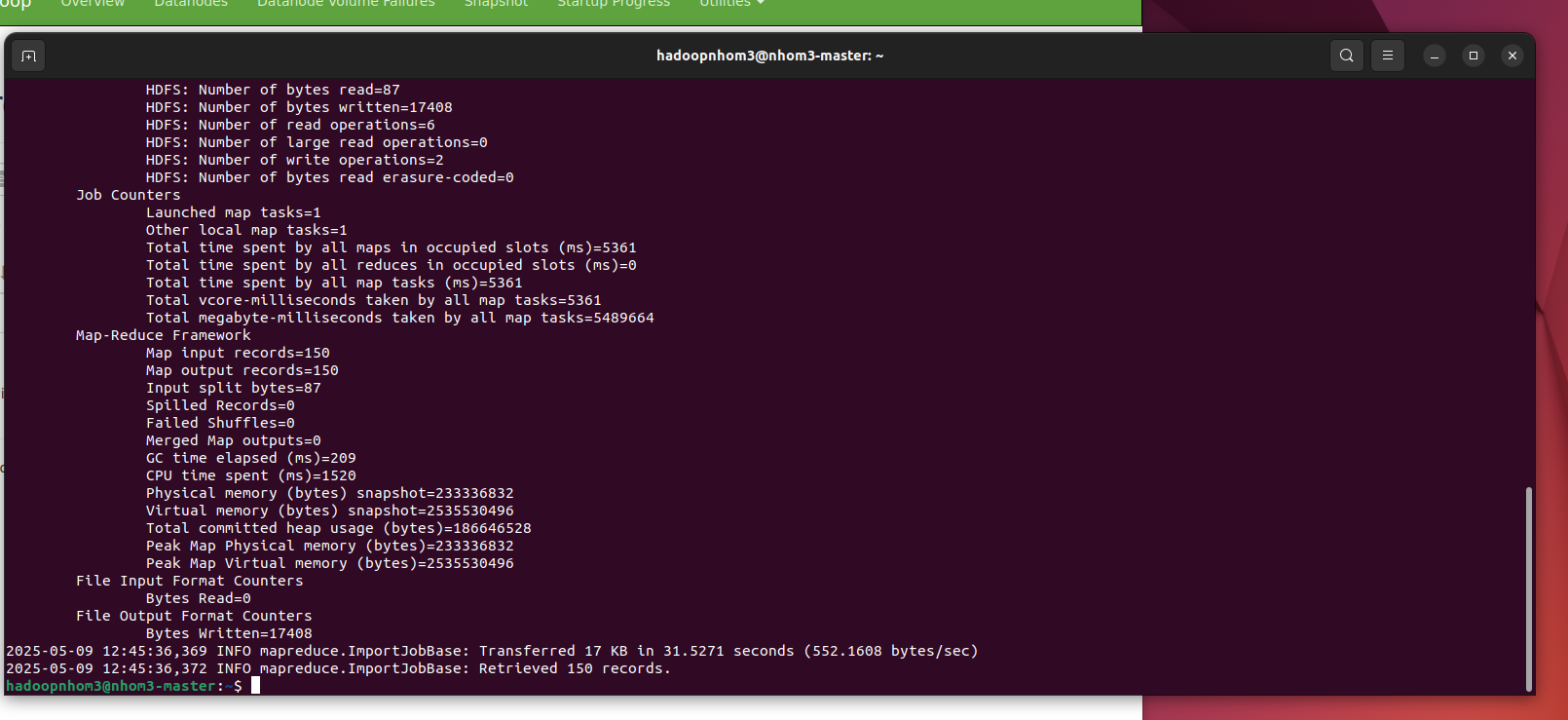
--table QUANAN \

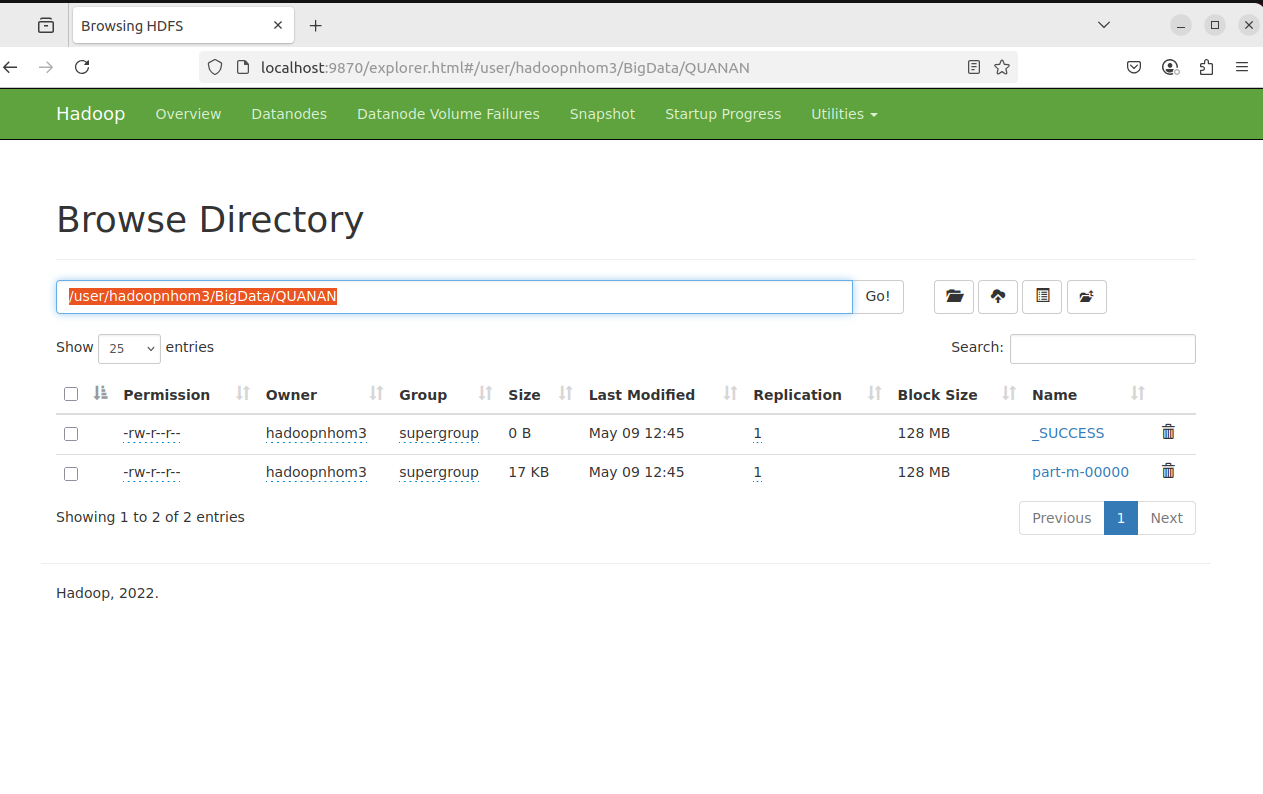
--target-dir /user/hadoopnhom3/BigData/QUANAN \

--fields-terminated-by '\t' \

--lines-terminated-by '\n' \

-m 1





**Xuất dữ liệu bảng DOAN**

sqoop import \

--connect jdbc:mysql://localhost:3306/QuanLyQuanAn \

--driver com.mysql.jdbc.Driver \

--username sqoopdb \

-P \

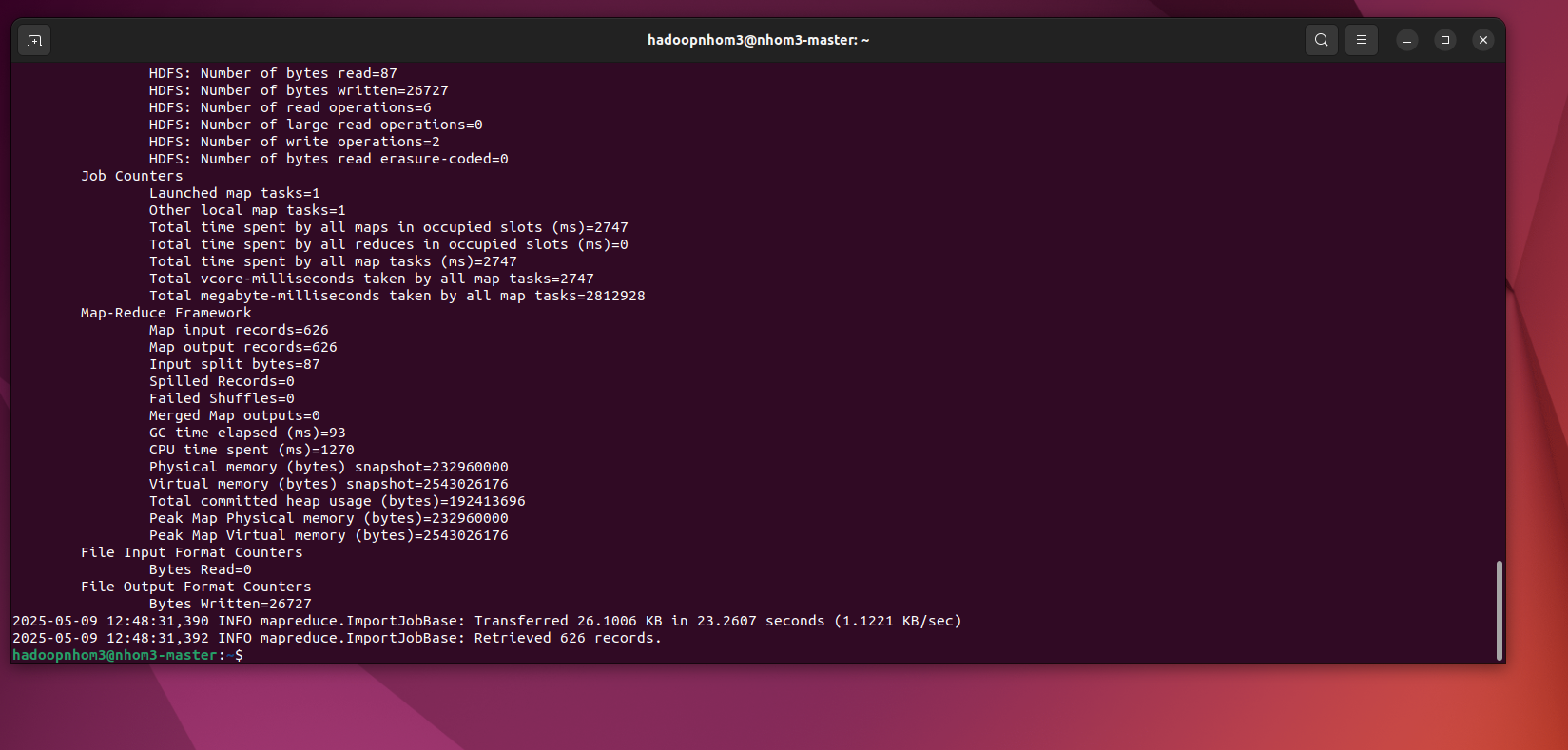
--table DOAN \

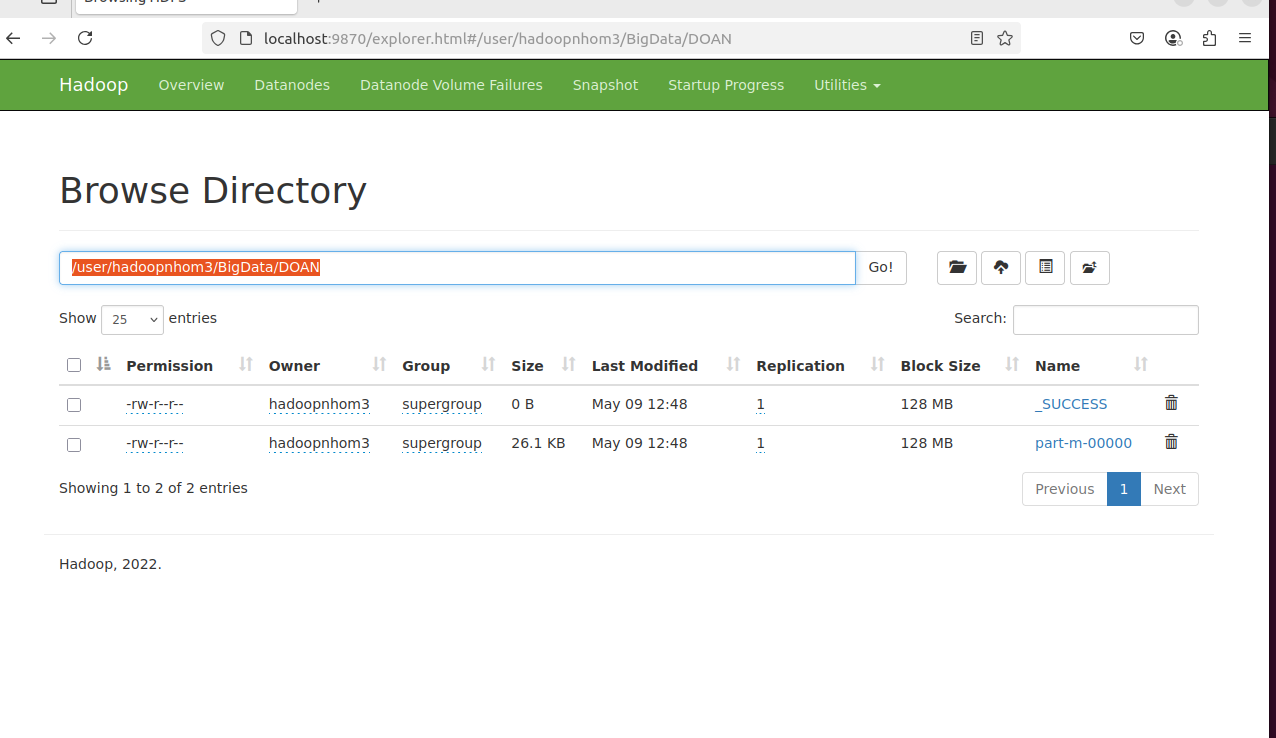
--target-dir /user/hadoopnhom3/BigData/DOAN \

--fields-terminated-by '\t' \

--lines-terminated-by '\n' \

-m 1





**Chương trình Mapreduce tìm 10 đồ ăn có giá cao nhất**

MapperDoAnMaxPrice.py

|  |
| --- |
| #!/usr/bin/env python3  import sys  for line in sys.stdin:  line = line.strip()  fields = line.split('\t')  if len(fields) != 4:  continue  try:  item = fields[2]  price = int(fields[3])  # Dùng price là key để sắp xếp dễ  print(f"{item}\t{price}")  except ValueError:  continue |

ReducerDoAnMaxPrice.py

|  |
| --- |
| #!/usr/bin/env python3  import sys  import heapq  top\_items = []  for line in sys.stdin:  line = line.strip()  parts = line.split('\t')  if len(parts) != 2:  continue  try:  item = parts[0]  price = int(parts[1])  heapq.heappush(top\_items, (price, item))  if len(top\_items) > 10:  heapq.heappop(top\_items)  except ValueError:  continue  for price, item in sorted(top\_items, reverse=True):  print(f"{item}\t{price}") |

Chạy chương trình

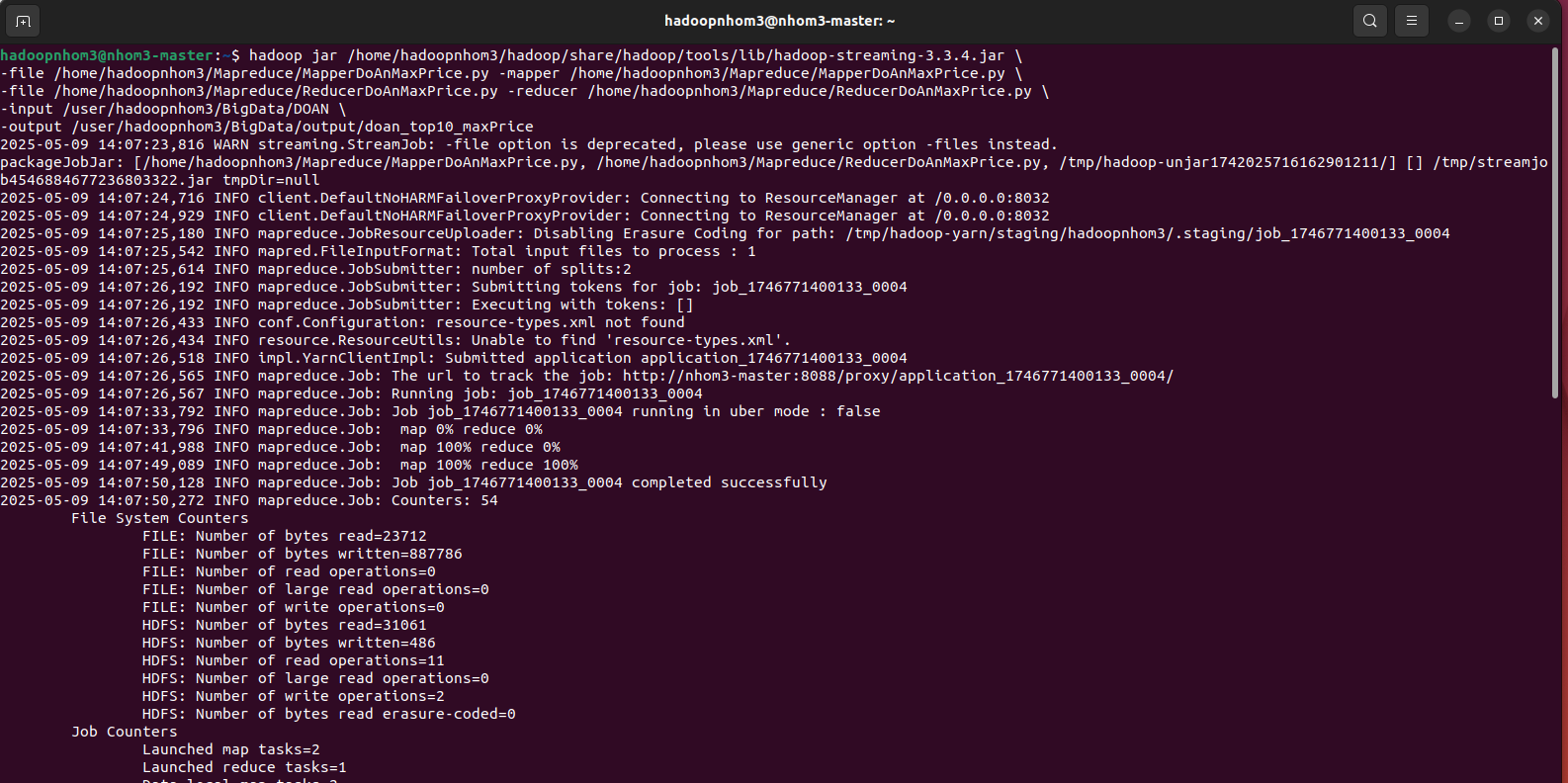
$ hadoop jar /home/hadoopnhom3/hadoop/share/hadoop/tools/lib/hadoop-streaming-3.3.4.jar \

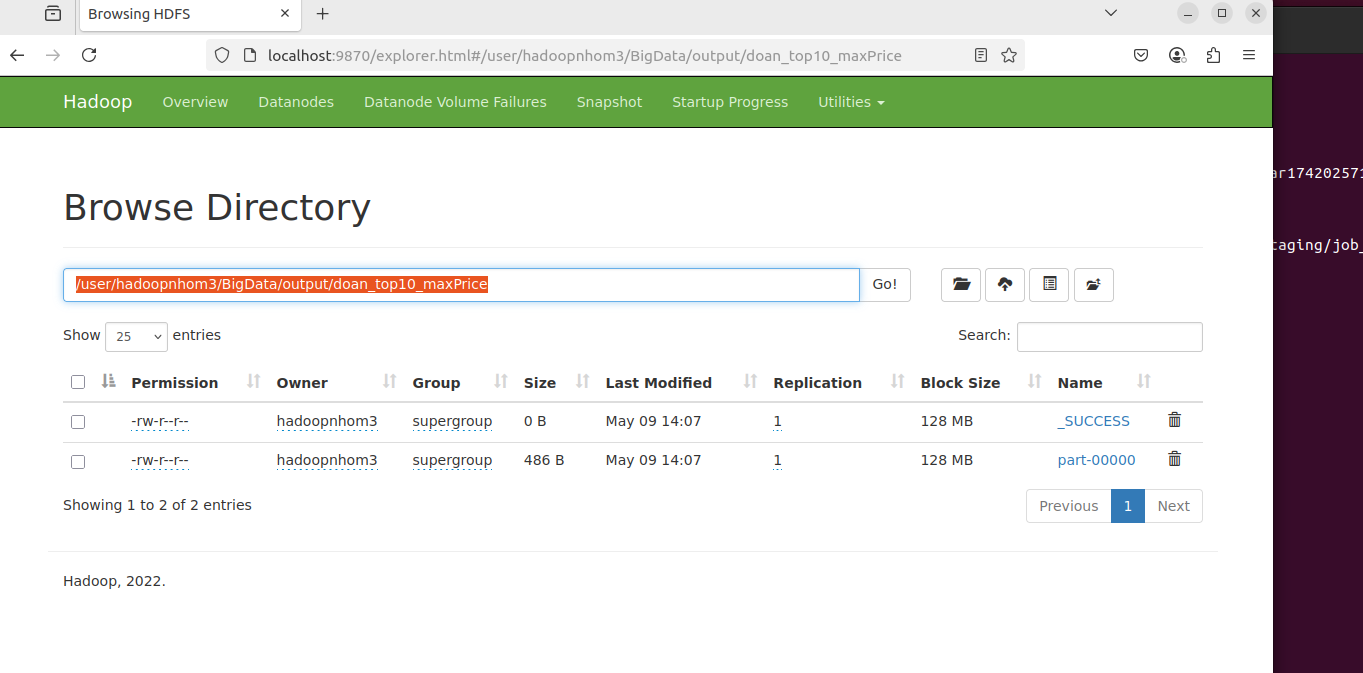
-file /home/hadoopnhom3/Mapreduce/MapperDoAnMaxPrice.py -mapper /home/hadoopnhom3/Mapreduce/MapperDoAnMaxPrice.py \

-file /home/hadoopnhom3/Mapreduce/ReducerDoAnMaxPrice.py -reducer /home/hadoopnhom3/Mapreduce/ReducerDoAnMaxPrice.py \

-input /user/hadoopnhom3/BigData/DOAN \

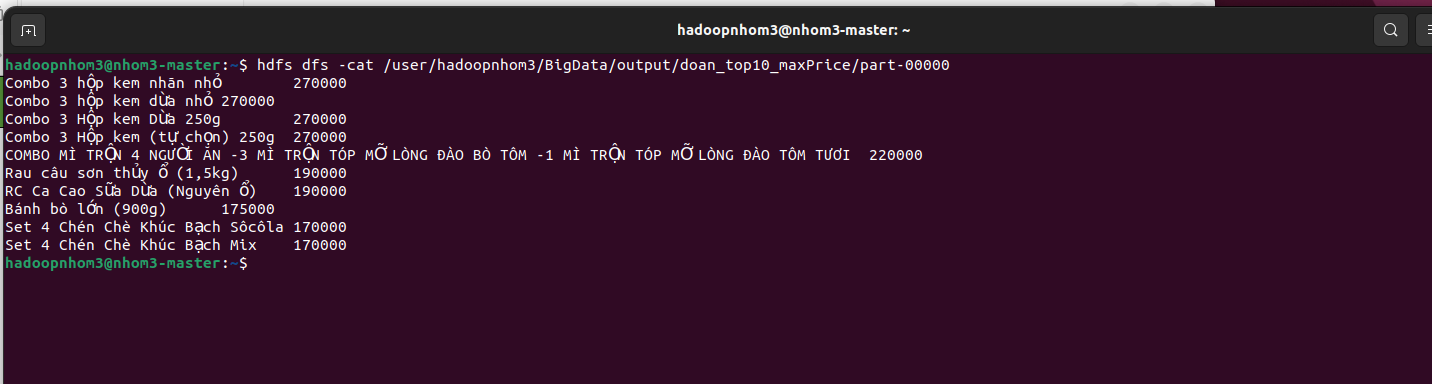
-output /user/hadoopnhom3/BigData/output/doan\_top10\_maxPrice





Hiển thị kết quả

$ hdfs dfs -cat /user/hadoopnhom3/BigData/output/doan\_top10\_maxPrice/part-00000



**Chương trình Mapreduce tìm 10 đồ ăn có giá thấp nhất**

MapperDoAnMinPrice.py

|  |
| --- |
| #!/usr/bin/env python3  import sys  for line in sys.stdin:  line = line.strip()  fields = line.split('\t')  if len(fields) != 4:  continue  try:  item = fields[2]  price = int(fields[3])  print(f"{item}\t{price}")  except ValueError:  continue |

ReducerDoAnMinPrice.py

|  |
| --- |
| #!/usr/bin/env python3  import sys  import heapq  top\_items = []  for line in sys.stdin:  line = line.strip()  parts = line.split('\t')  if len(parts) != 2:  continue  try:  item = parts[0]  price = int(parts[1])  heapq.heappush(top\_items, (-price, item))  if len(top\_items) > 10:  heapq.heappop(top\_items)  except ValueError:  continue  for neg\_price, item in sorted(top\_items, reverse=True):  print(f"{item}\t{-neg\_price}") |

Chạy chương trình

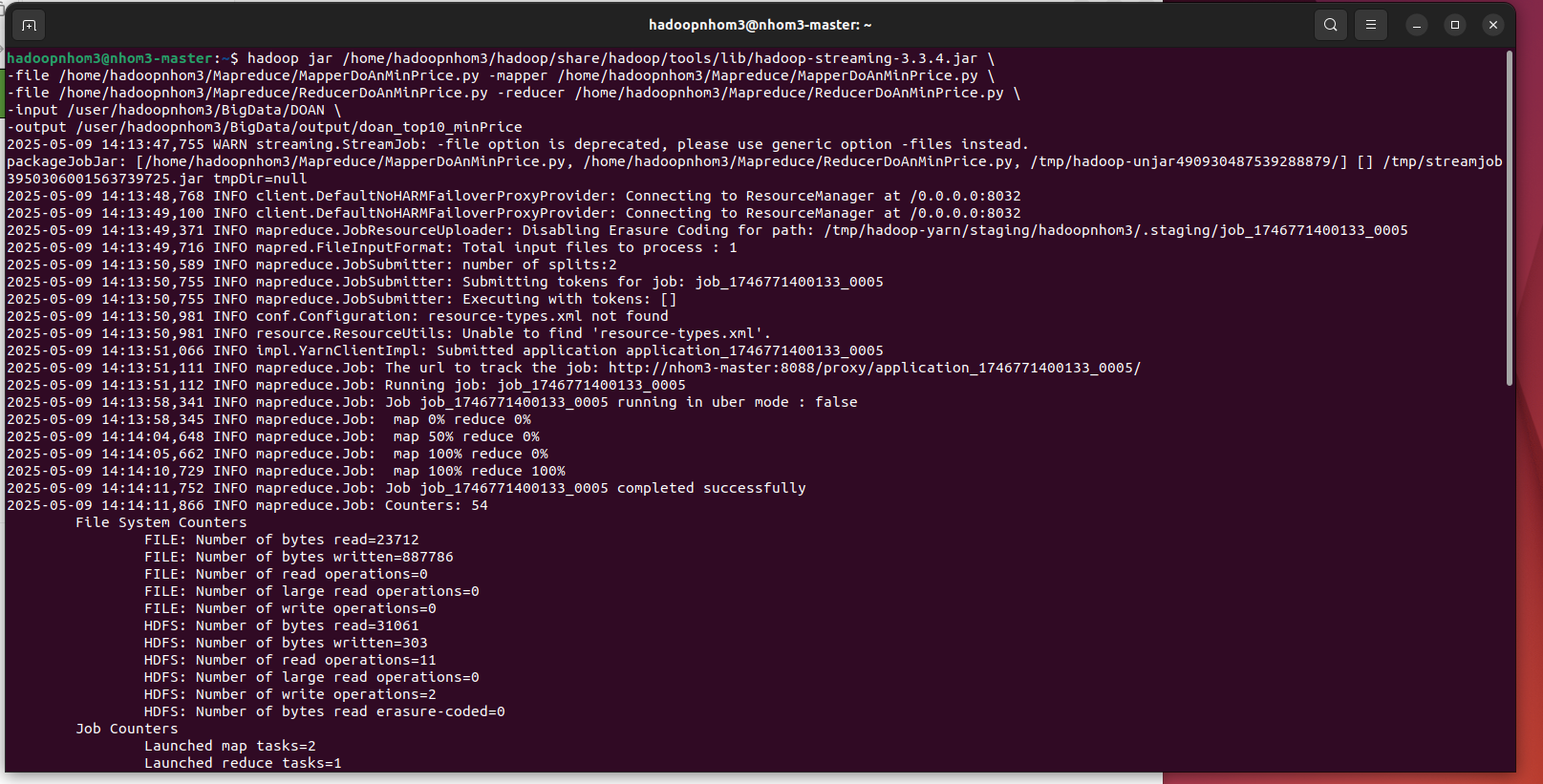
$ hadoop jar /home/hadoopnhom3/hadoop/share/hadoop/tools/lib/hadoop-streaming-3.3.4.jar \

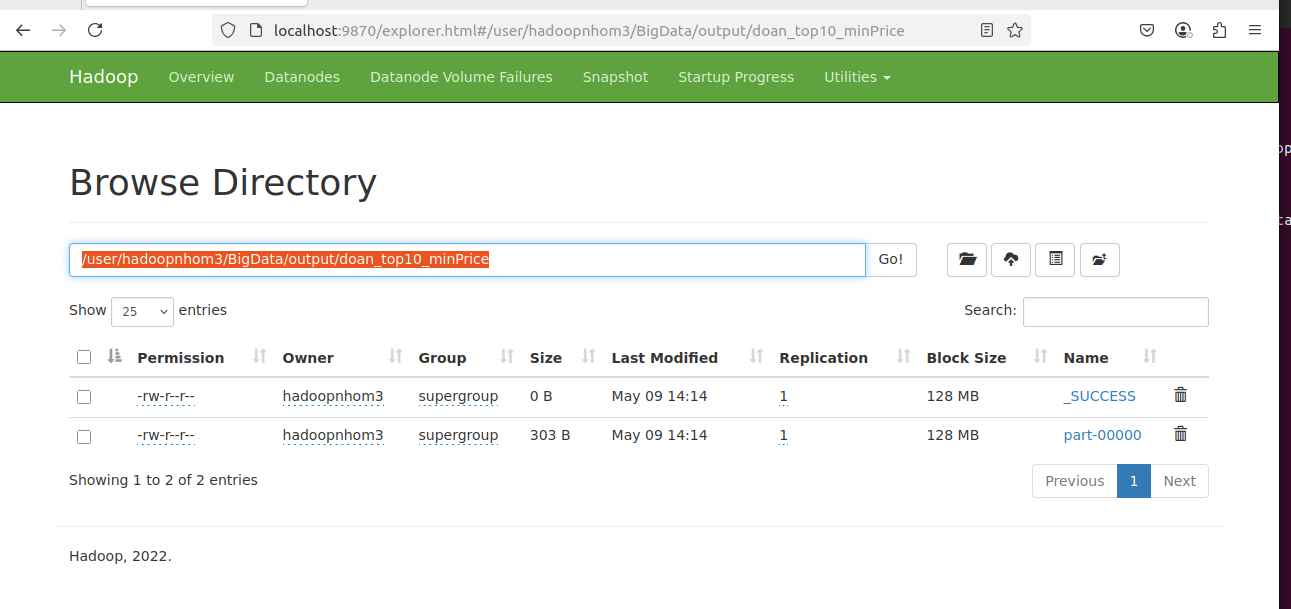
-file /home/hadoopnhom3/Mapreduce/MapperDoAnMinPrice.py -mapper /home/hadoopnhom3/Mapreduce/MapperDoAnMinPrice.py \

-file /home/hadoopnhom3/Mapreduce/ReducerDoAnMinPrice.py -reducer /home/hadoopnhom3/Mapreduce/ReducerDoAnMinPrice.py \

-input /user/hadoopnhom3/BigData/DOAN \

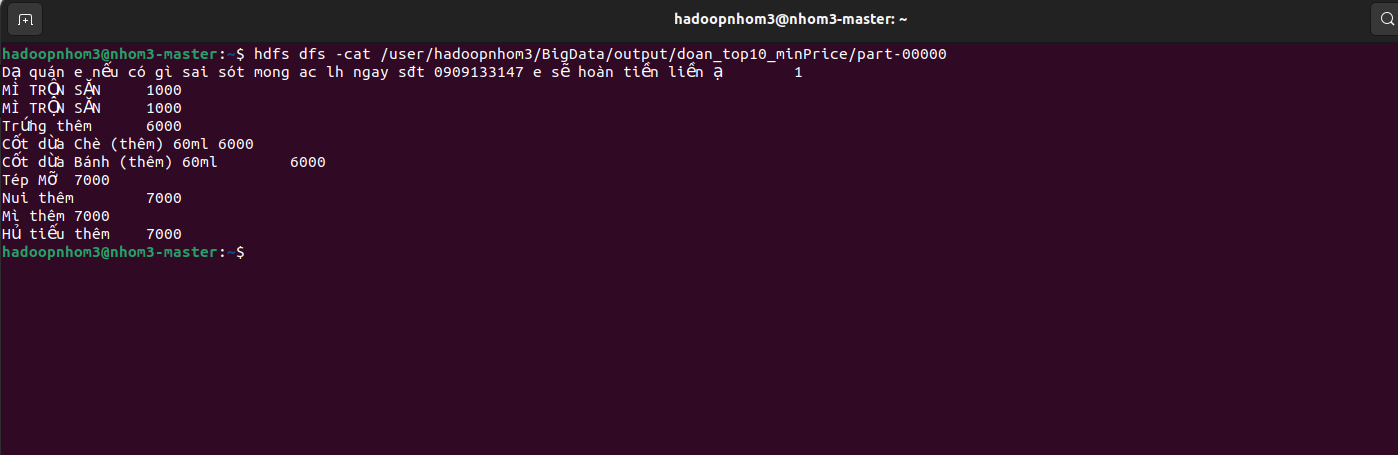
-output /user/hadoopnhom3/BigData/output/doan\_top10\_minPrice





Hiển thị kết quả

$ hdfs dfs -cat /user/hadoopnhom3/BigData/output/doan\_top10\_minPrice/part-00000



**Chương trình Mapreduce tìm 10 quán ăn có đánh giá cao nhất**

MapperQuanAnMaxRating.py

|  |
| --- |
| #!/usr/bin/env python3  import sys  for line in sys.stdin:  line = line.strip()  fields = line.split('\t')  if len(fields) != 4:  continue  try:  tenquan = fields[1]  rating = float(fields[3])  print(f"{tenquan}\t{rating}")  except ValueError:  continue |

ReducerQuanAnMaxRating.py

|  |
| --- |
| #!/usr/bin/env python3  import sys  import heapq  top\_quans = []  for line in sys.stdin:  line = line.strip()  parts = line.split('\t')  if len(parts) != 2:  continue  try:  tenquan = parts[0]  rating = float(parts[1])  heapq.heappush(top\_quans, (rating, tenquan))  if len(top\_quans) > 10:  heapq.heappop(top\_quans)  except ValueError:  continue  for rating, tenquan in sorted(top\_quans, reverse=True):  print(f"{tenquan}\t{rating}") |

Chạy chương trình

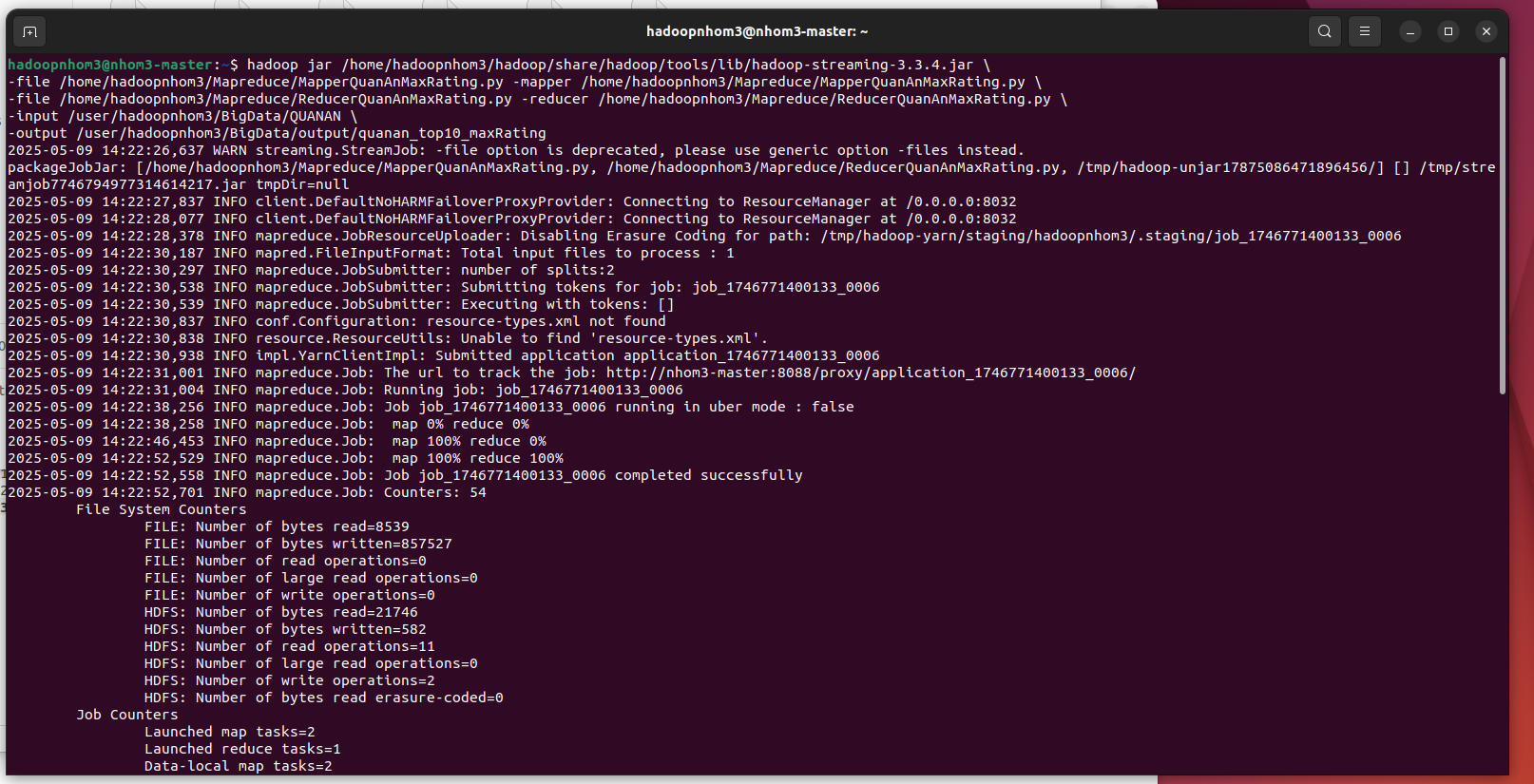
$ hadoop jar /home/hadoopnhom3/hadoop/share/hadoop/tools/lib/hadoop-streaming-3.3.4.jar \

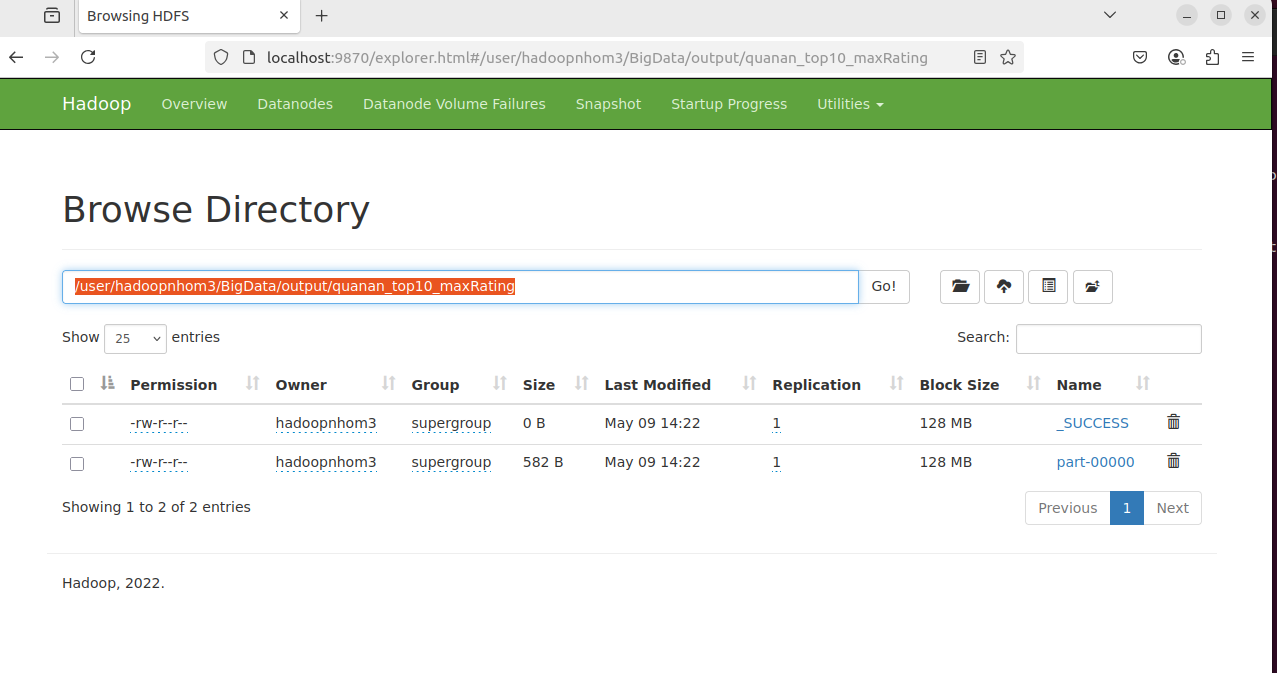
-file /home/hadoopnhom3/Mapreduce/MapperQuanAnMaxRating.py -mapper /home/hadoopnhom3/Mapreduce/MapperQuanAnMaxRating.py \

-file /home/hadoopnhom3/Mapreduce/ReducerQuanAnMaxRating.py -reducer /home/hadoopnhom3/Mapreduce/ReducerQuanAnMaxRating.py \

-input /user/hadoopnhom3/BigData/QUANAN \

-output /user/hadoopnhom3/BigData/output/quanan\_top10\_maxRating





Hiển thị kết quả

$ hdfs dfs -cat /user/hadoopnhom3/BigData/output/quanan\_top10\_maxRating/part-00000

