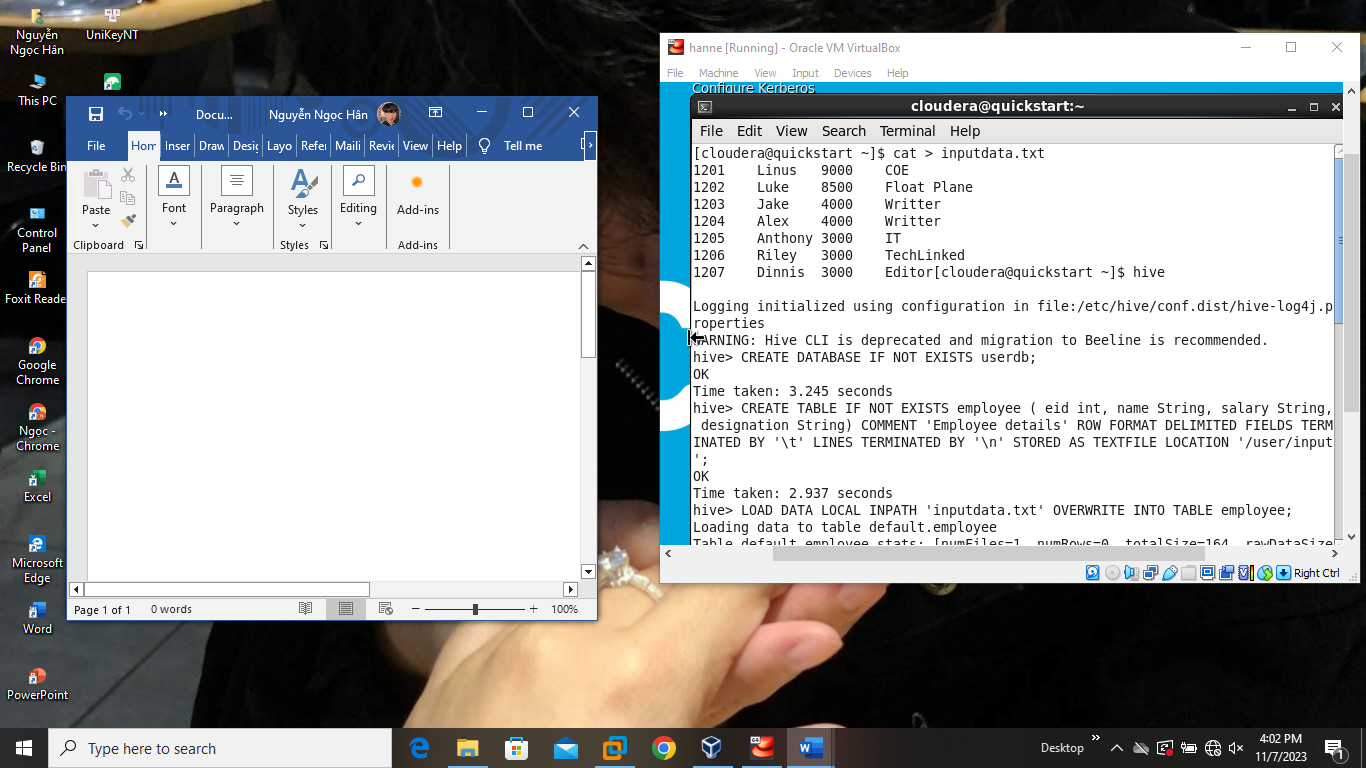
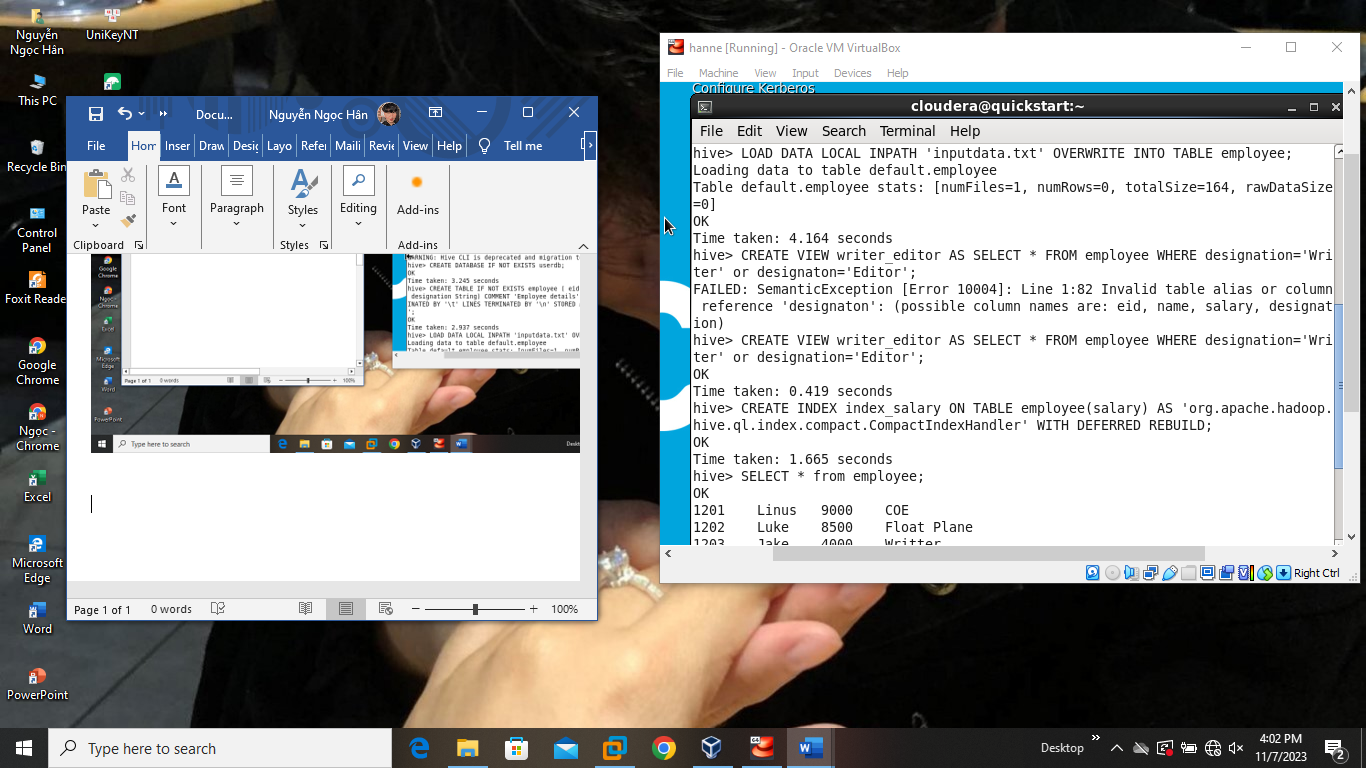
Bùi Thanh Tuyền

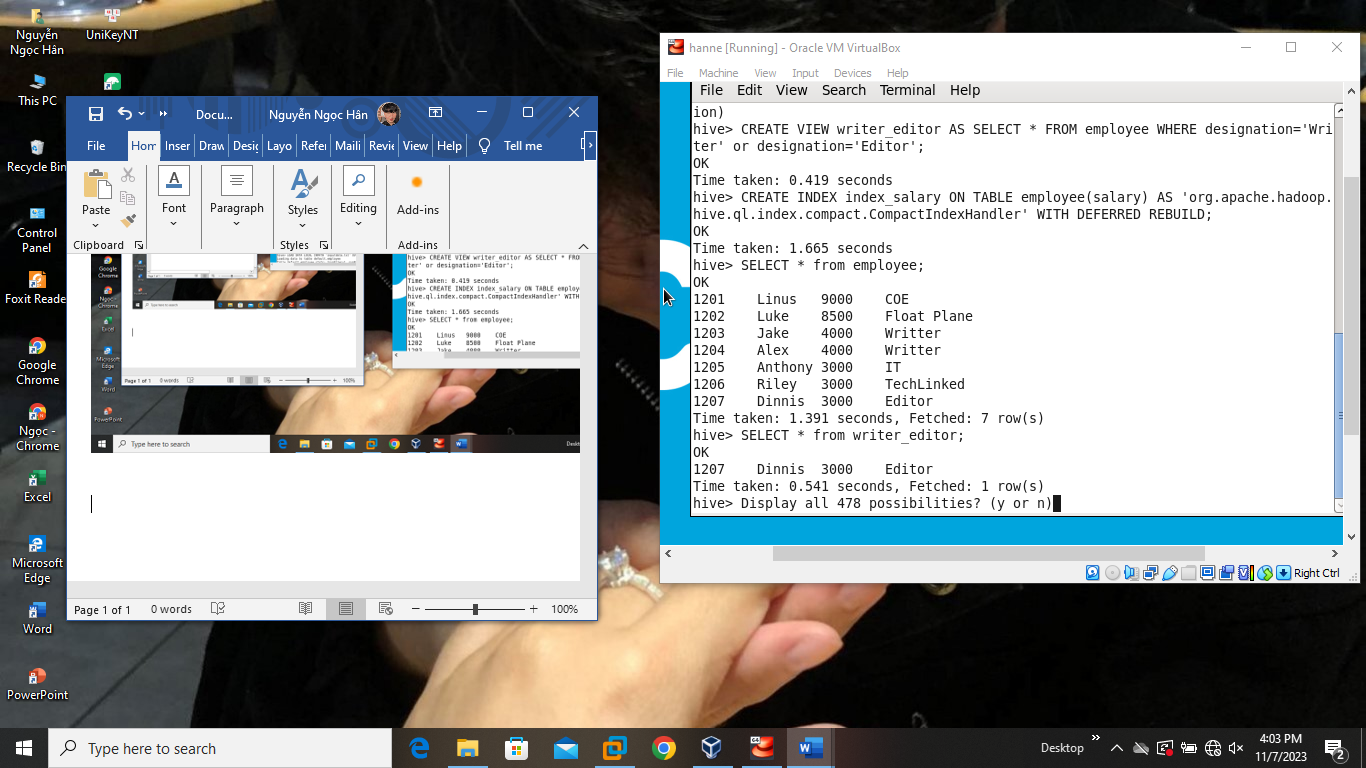
MSSV: 1050080040

10ĐH\_CNTT1

Lap8

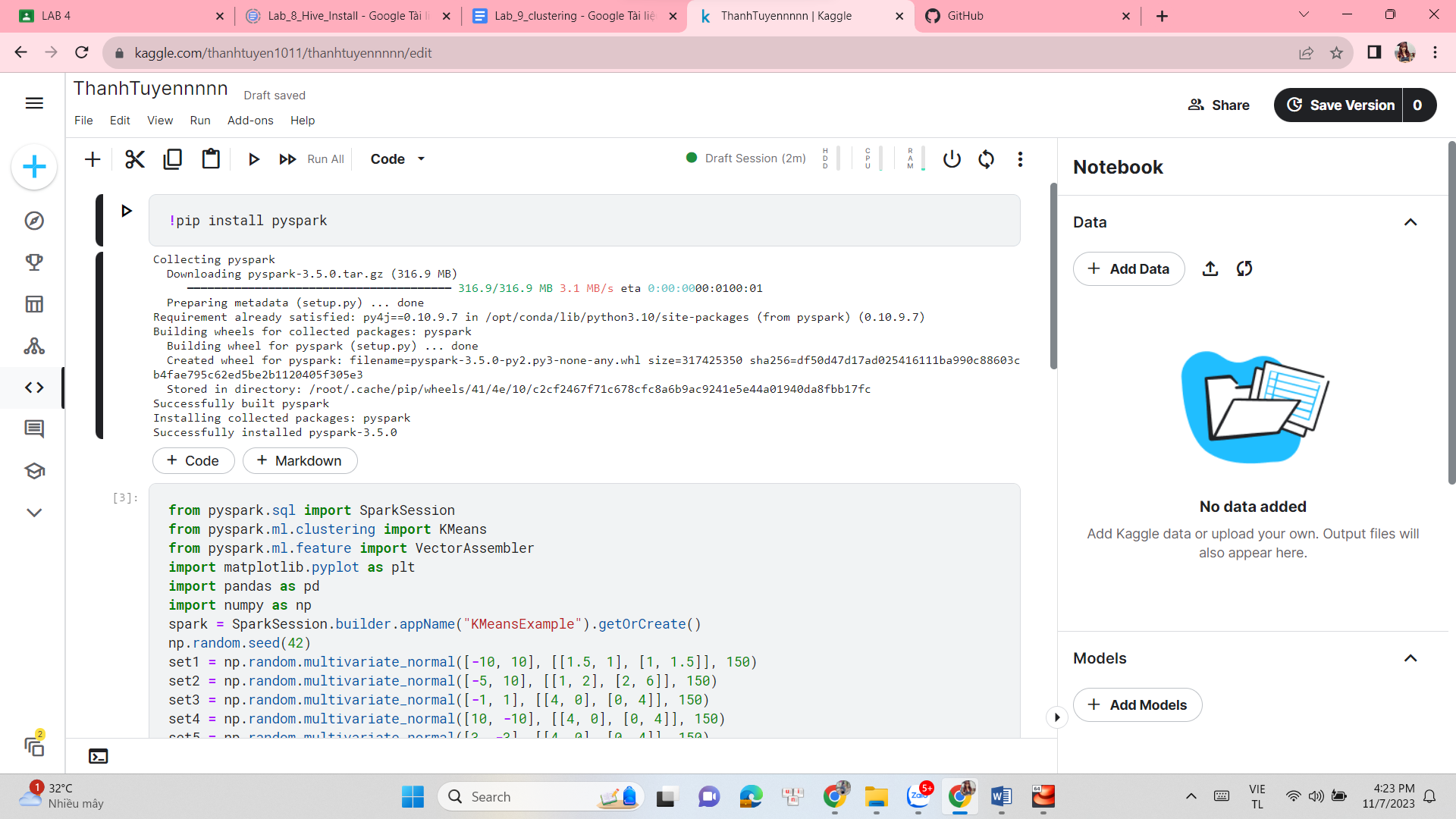






Lap9

!pip install pyspark



from pyspark.sql import SparkSession

from pyspark.ml.clustering import KMeans

from pyspark.ml.feature import VectorAssembler

import matplotlib.pyplot as plt

import pandas as pd

import numpy as np

spark = SparkSession.builder.appName("KMeansExample").getOrCreate()

np.random.seed(42)

set1 = np.random.multivariate\_normal([-10, 10], [[1.5, 1], [1, 1.5]], 150)

set2 = np.random.multivariate\_normal([-5, 10], [[1, 2], [2, 6]], 150)

set3 = np.random.multivariate\_normal([-1, 1], [[4, 0], [0, 4]], 150)

set4 = np.random.multivariate\_normal([10, -10], [[4, 0], [0, 4]], 150)

set5 = np.random.multivariate\_normal([3, -3], [[4, 0], [0, 4]], 150)

data = np.concatenate((set1, set2, set3, set4, set5))

columns = ["X1", "X2"]

df = pd.DataFrame(data, columns=columns)

df["cluster"] = np.repeat(range(1, 6), 150)

data = spark.createDataFrame(df)

vec\_assembler = VectorAssembler(inputCols=["X1", "X2"], outputCol="features")

data = vec\_assembler.transform(data)

K = 5

kmeans = KMeans().setK(K).setSeed(1)

model = kmeans.fit(data)

centroids = model.clusterCenters()

predictions = model.transform(data)

data\_with\_clusters = predictions.withColumn("cluster", predictions["prediction"])

data\_with\_clusters = data\_with\_clusters.withColumn("isCentroid", data\_with\_clusters["prediction"] == data\_with\_clusters["cluster"])

data\_to\_plot = data\_with\_clusters.toPandas()

centroids = np.array(centroids)

plt.scatter(data\_to\_plot["X1"], data\_to\_plot["X2"], c=data\_to\_plot["cluster"], s=50, alpha=0.7)

plt.scatter(centroids[:, 0], centroids[:, 1], c="red", s=200, marker="x", label="Centroids")

plt.legend()

plt.xlabel("X1")

plt.ylabel("X2")

plt.title("K-Means Clustering")

plt.show()

