```
!wget https://archive.apache.org/dist/spark/spark-3.4.0/spark-3.4.0-bin-hadoop3.tgz
```

```
--2025-11-20 04:04:07--  https://archive.apache.org/dist/spark/spark-3.4.0/spark-3.4.0-bin-hadoop3.tgz
Resolving archive.apache.org (archive.apache.org)... 65.108.204.189, 2a01:4f9:1a:a084::2
Connecting to archive.apache.org (archive.apache.org)|65.108.204.189|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 388407094 (370M) [application/x-gzip]
Saving to: ‘spark-3.4.0-bin-hadoop3.tgz.1’

spark-3.4.0-bin-had 100%[===================>] 370.41M   338KB/s    in 26m 2s

2025-11-20 04:30:10 (243 KB/s) - ‘spark-3.4.0-bin-hadoop3.tgz.1’ saved [388407094/388407094]
```

```
!apt-get install openjdk-17-jdk-headless -qq
```

```
!tar -xzf spark-3.4.0-bin-hadoop3.tgz
```

```
!pip install -q findspark
```

```
import findspark
findspark.init("/content/spark-3.4.0-bin-hadoop3")

from pyspark.sql import SparkSession
spark = SparkSession.builder.appName("BDA Lab").getOrCreate()
print("Spark Ready!")
```

```
Spark Ready!
```

```
from pyspark.ml.clustering import KMeans
from pyspark.ml.linalg import Vectors
```

```
data = [(1.0,1.0), (2.0,1.0), (1.0,2.0),
        (8.0,8.0), (9.0,8.0), (8.0,9.0)]
```

```
df = spark.createDataFrame([(Vectors.dense(x),) for x in data],
                           ["features"])
df.show()
```

```
+---------+
| features|
+---------+
|[1.0,1.0]|
|[2.0,1.0]|
|[1.0,2.0]|
|[8.0,8.0]|
|[9.0,8.0]|
|[8.0,9.0]|
+---------+
```

```
kmeans = KMeans(k=2)
model = kmeans.fit(df)
```

```
print("Cluster Centers:")
for c in model.clusterCenters():
    print(c)
```

```
Cluster Centers:
[8.33333333 8.33333333]
[1.33333333 1.33333333]
```

```
pred = model.transform(df)
pred.show()
```

```
+---------+----------+
| features|prediction|
+---------+----------+
|[1.0,1.0]|         1|
|[2.0,1.0]|         1|
|[1.0,2.0]|         1|
|[8.0,8.0]|         0|
|[9.0,8.0]|         0|
|[8.0,9.0]|         0|
+---------+----------+
```

```
from pyspark.ml.clustering import KMeans
from pyspark.ml.linalg import Vectors

data = [(1.0,1.0), (2.0,1.0), (1.0,2.0),
        (8.0,8.0), (9.0,8.0), (8.0,9.0)]

df = spark.createDataFrame([(Vectors.dense(x),) for x in data], ["features"])

kmeans = KMeans(k=2)
model = kmeans.fit(df)

print("Cluster Centers:")
for c in model.clusterCenters():
    print(c)

model.transform(df).show()
```

```
Cluster Centers:
[8.33333333 8.33333333]
[1.33333333 1.33333333]
+---------+----------+
| features|prediction|
+---------+----------+
|[1.0,1.0]|         1|
|[2.0,1.0]|         1|
|[1.0,2.0]|         1|
|[8.0,8.0]|         0|
|[9.0,8.0]|         0|
|[8.0,9.0]|         0|
+---------+----------+
```