**Hbase Assignment2\_Anup\_Patil  
  
Access Tables through python**  
  
A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

import happybase

try:

connection = happybase.Connection('10.1.1.204', 9090)

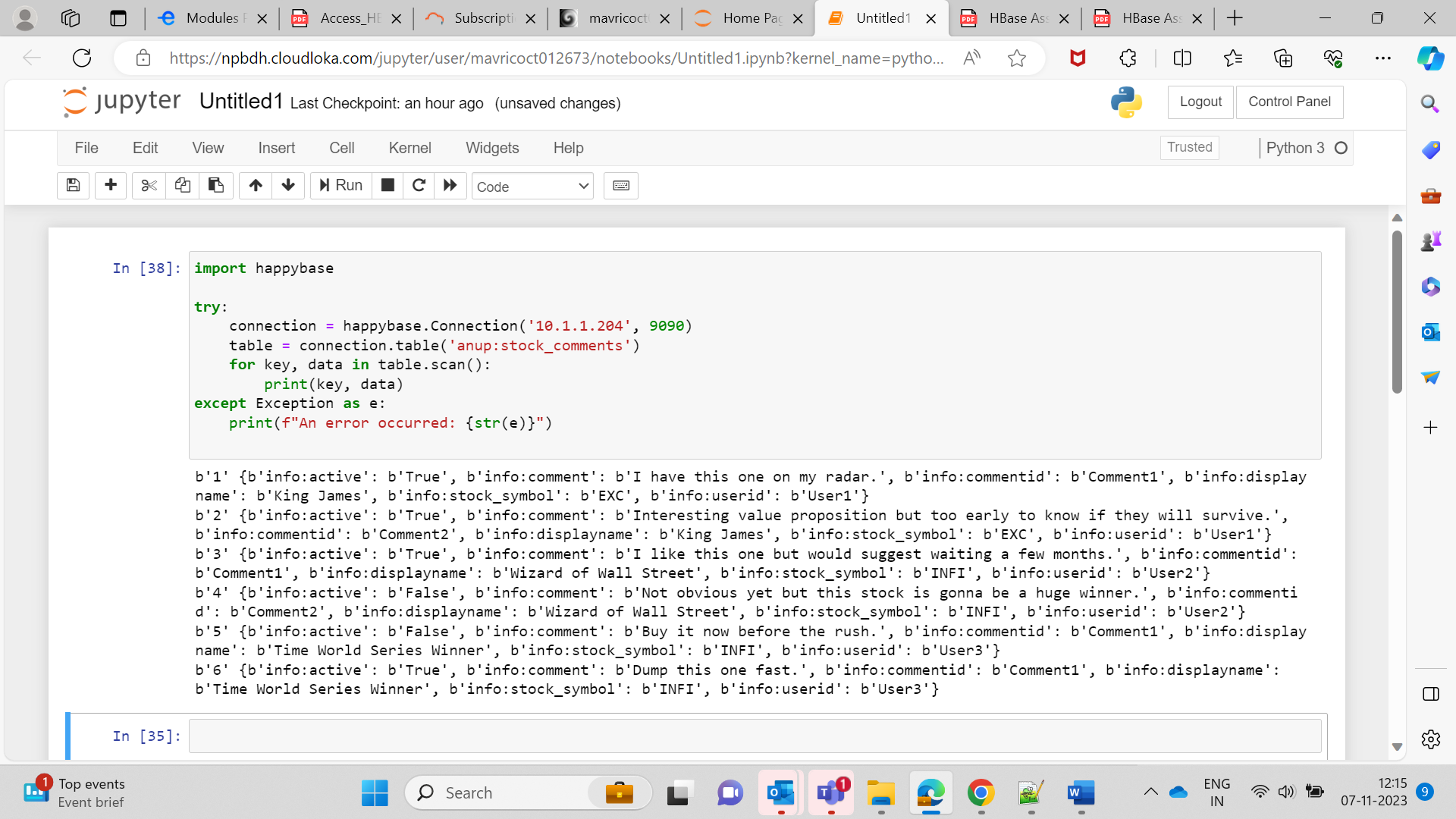
table = connection.table('anup:stock\_comments')

for key, data in table.scan():

print(key, data)

except Exception as e:

print(f"An error occurred: {str(e)}")  
  
  
  
**Queries First Tables using python-**

  
  
**Q1.** **Various comments by all users  
  
import happybase**

**try:**

**connection = happybase.Connection('10.1.1.204', 9090)**

**table = connection.table('anup:stock\_comments')**

**print("Various comments by all users")**

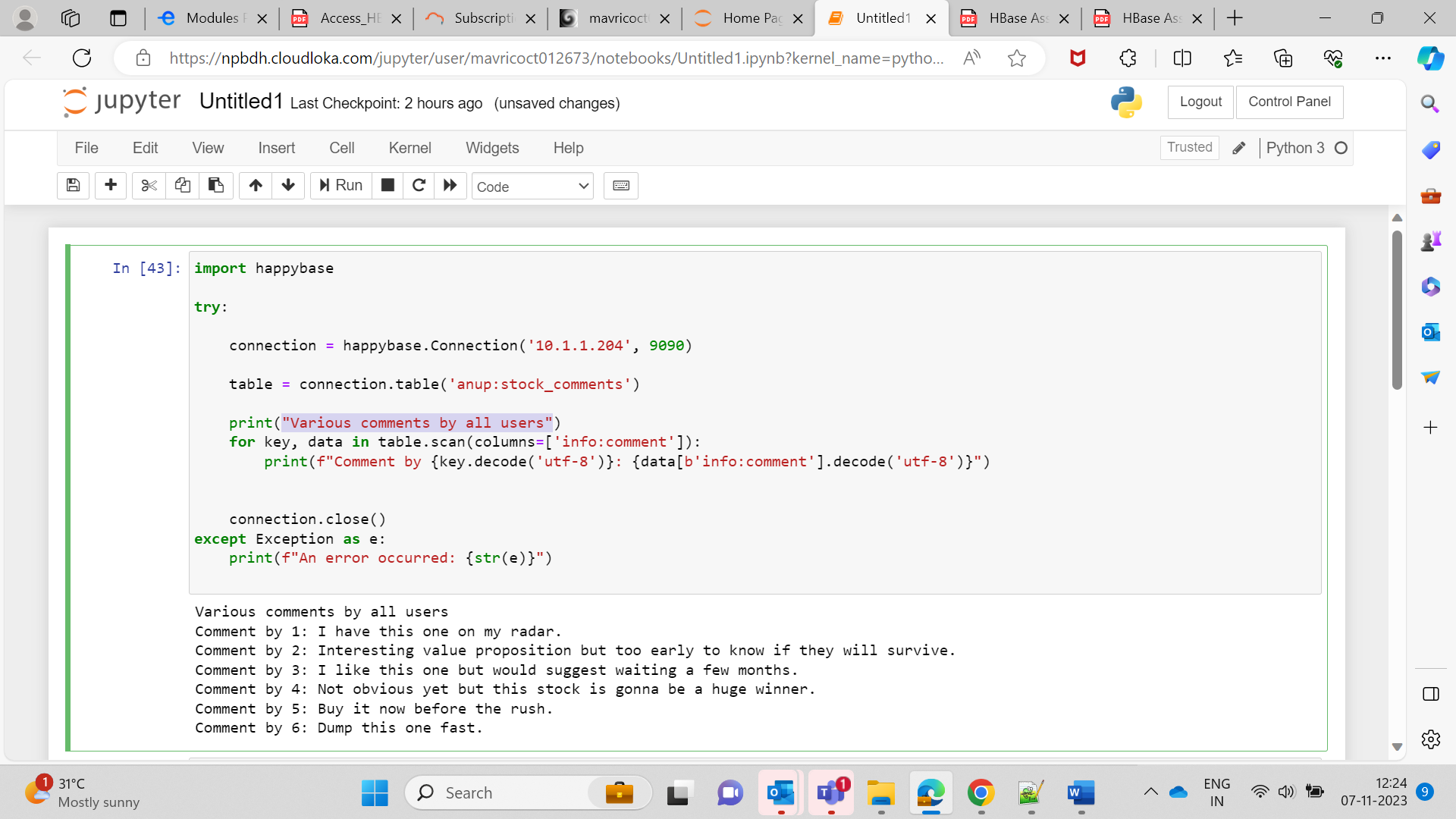
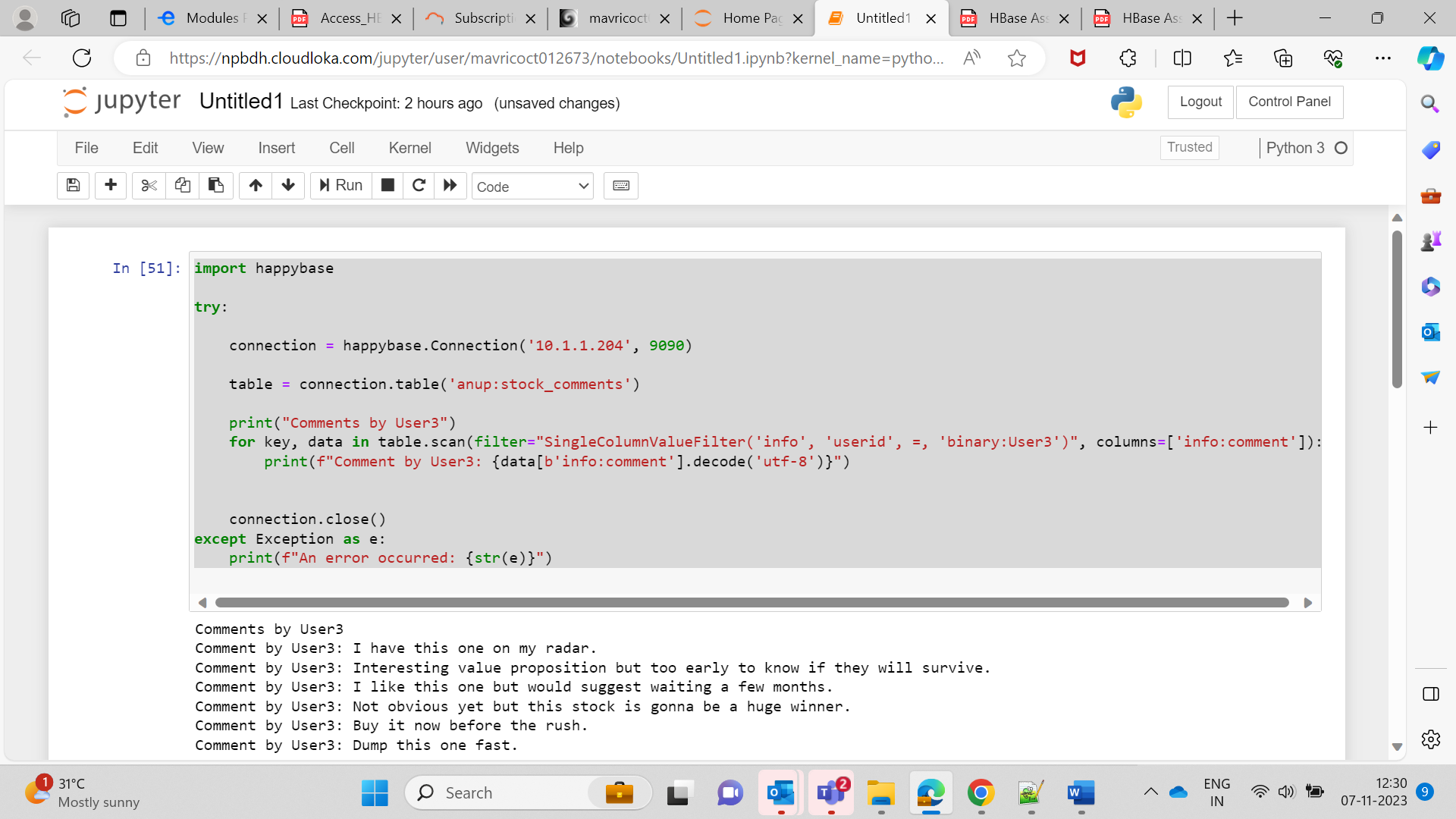
**for key, data in table.scan(columns=['info:comment']):**

**print(f"Comment by {key.decode('utf-8')}: {data[b'info:comment'].decode('utf-8')}")**

**connection.close()**

**except Exception as e:**

**print(f"An error occurred: {str(e)}")**

  
  
Q2. Comments by User3  


import happybase

try:

connection = happybase.Connection('10.1.1.204', 9090)

table = connection.table('anup:stock\_comments')

print("Comments by User3")

for key, data in table.scan(filter="SingleColumnValueFilter('info', 'userid', =, 'binary:User3')", columns=['info:comment']):

print(f"Comment by User3: {data[b'info:comment'].decode('utf-8')}")

connection.close()

except Exception as e:

print(f"An error occurred: {str(e)}")  
  
  
Q3. Query 3: Users with inactive comments  
  
A screenshot of a computer

Description automatically generated

import happybase

try:

connection = happybase.Connection('10.1.1.204', 9090)

table = connection.table('anup:stock\_comments')

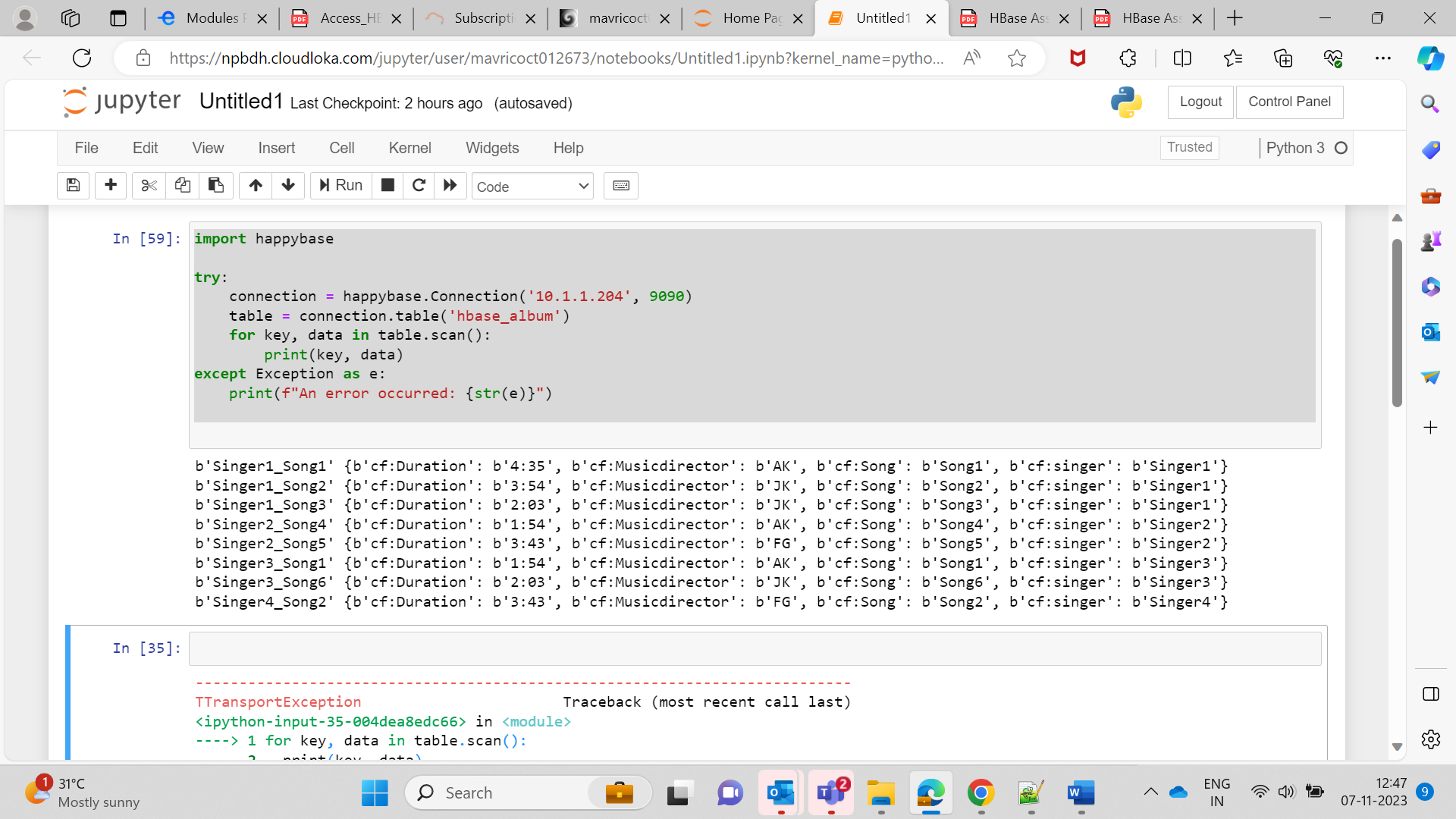
print("Users with inactive comments")

for key, data in table.scan(filter="SingleColumnValueFilter('info', 'active', =, 'binary:False')", columns=['info:userid']):

print(f"User with inactive comment: {data[b'info:userid'].decode('utf-8')}")

connection.close()

except Exception as e:

print(f"An error occurred: {str(e)}")  
  
  
**2 Access Album table Using Python**import happybase

try:

connection = happybase.Connection('10.1.1.204', 9090)

table = connection.table('hbase\_album')

for key, data in table.scan():

print(key, data)

except Exception as e:

print(f"An error occurred: {str(e)}")  
  
  
**1. Get all rows from the table"**  
  
A screenshot of a computer

Description automatically generated

import happybase

try:

connection = happybase.Connection('10.1.1.204', 9090)

table = connection.table('hbase\_album')

print("Get all rows from the table")

for key, data in table.scan():

print(key, data)

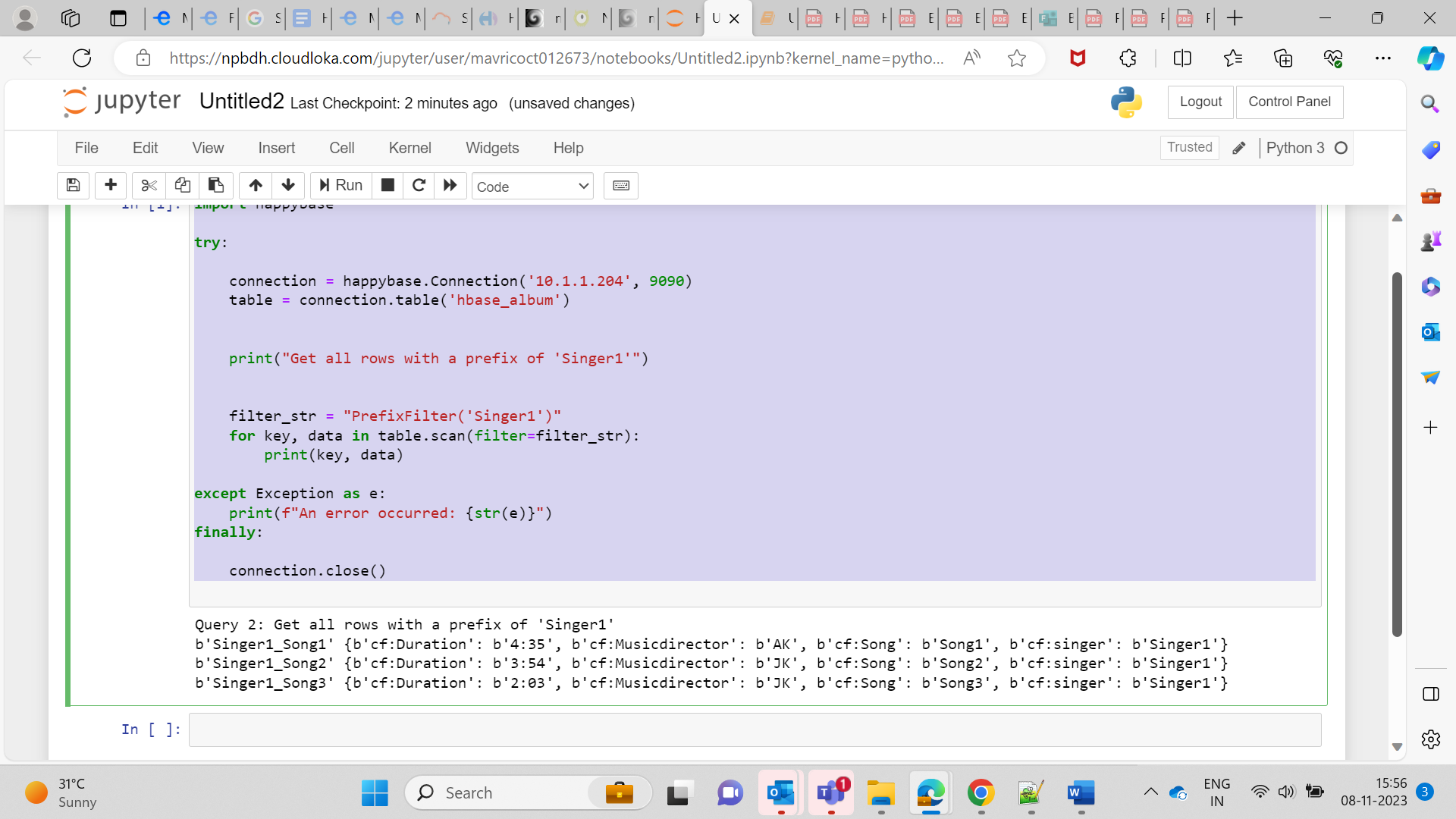
except Exception as e:

print(f"An error occurred: {str(e)}")

finally:

#

connection.close()

**Q2 Query- Get all rows with a prefix of 'Singer1'**  
  
  
  
  
  
  
import happybase

try:

connection = happybase.Connection('10.1.1.204', 9090)

table = connection.table('hbase\_album')

print("Get all rows with a prefix of 'Singer1'")

filter\_str = "PrefixFilter('Singer1')"

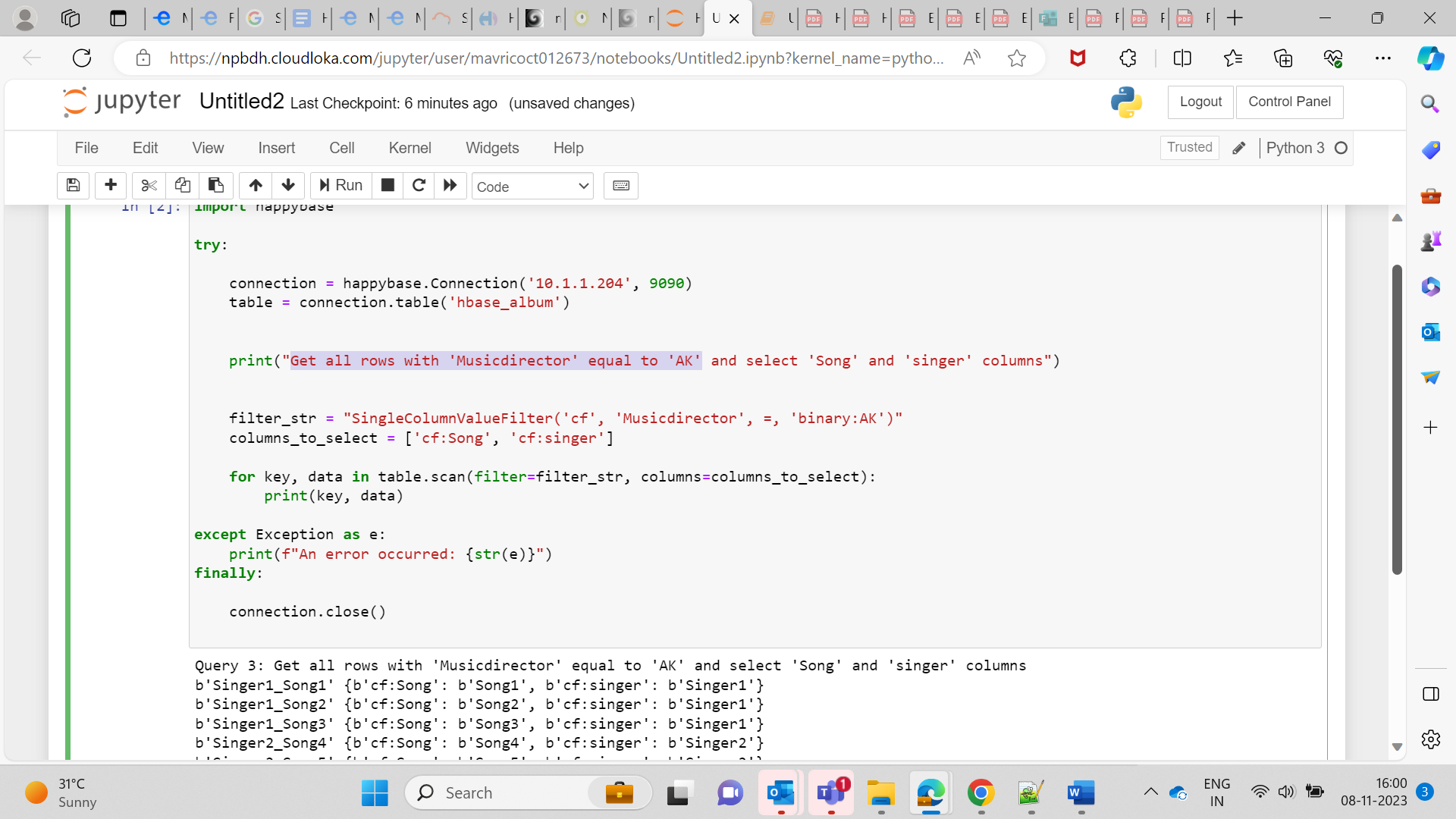
for key, data in table.scan(filter=filter\_str):

print(key, data)

except Exception as e:

print(f"An error occurred: {str(e)}")

finally:

connection.close()  
  
  
**Q3. Get all rows with 'Musicdirector' equal to 'AK'**

**import happybase**

**try:**

**connection = happybase.Connection('10.1.1.204', 9090)**

**table = connection.table('hbase\_album')**

**print("Get all rows with 'Musicdirector' equal to 'AK' and select 'Song' and 'singer' columns")**

**filter\_str = "SingleColumnValueFilter('cf', 'Musicdirector', =, 'binary:AK')"**

**columns\_to\_select = ['cf:Song', 'cf:singer']**

**for key, data in table.scan(filter=filter\_str, columns=columns\_to\_select):**

**print(key, data)**

**except Exception as e:**

**print(f"An error occurred: {str(e)}")**

**finally:**

**connection.close()**  
**Task3. Create Hbase Table using Python  
  
Mobiles Sales data Table**import happybase

try:

connection = happybase.Connection('10.1.1.204', 9090)

connection.open()

print("Connected to HBase")

# Create a 'mobile\_sales' table in the 'anup' namespace

connection.create\_table(

'anup:mobile\_sales',

{

'sales\_info': dict(), # Column family

}

)

print("Created 'mobile\_sales' table")

# Access the 'mobile\_sales' table within the 'anup' namespace

table = connection.table('anup:mobile\_sales')

# Insert sample data

sample\_data = {

'sale1': {

'sales\_info:device': 'iPhone',

'sales\_info:date': '2023-11-01',

'sales\_info:quantity': '10',

'sales\_info:revenue': '5000',

},

'sale2': {

'sales\_info:device': 'Samsung Galaxy',

'sales\_info:date': '2023-11-02',

'sales\_info:quantity': '8',

'sales\_info:revenue': '4000',

},

'sale3': {

'sales\_info:device': 'Google Pixel',

'sales\_info:date': '2023-11-03',

'sales\_info:quantity': '5',

'sales\_info:revenue': '2500',

},

}

with table.batch() as b:

for row\_key, data in sample\_data.items():

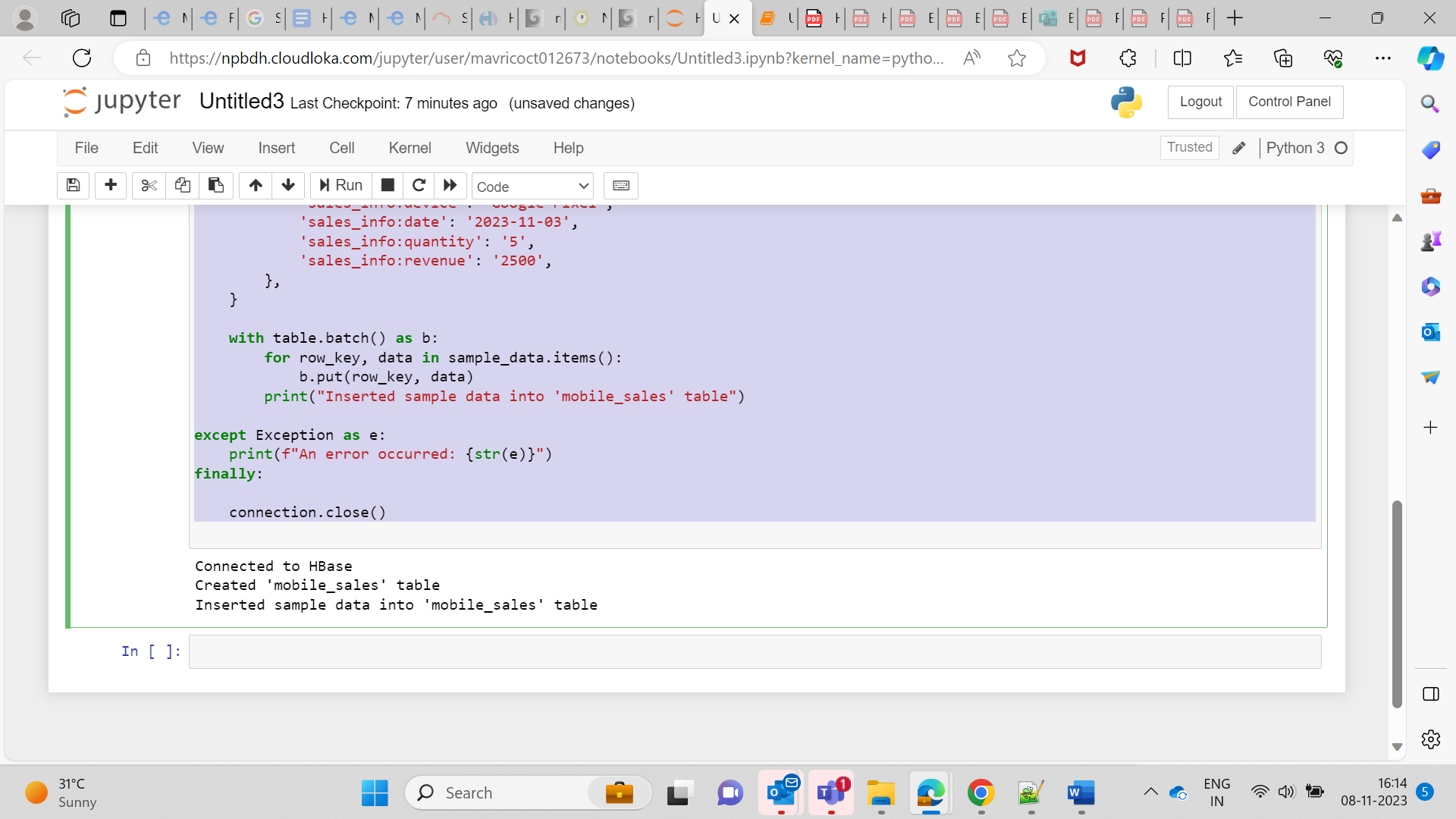
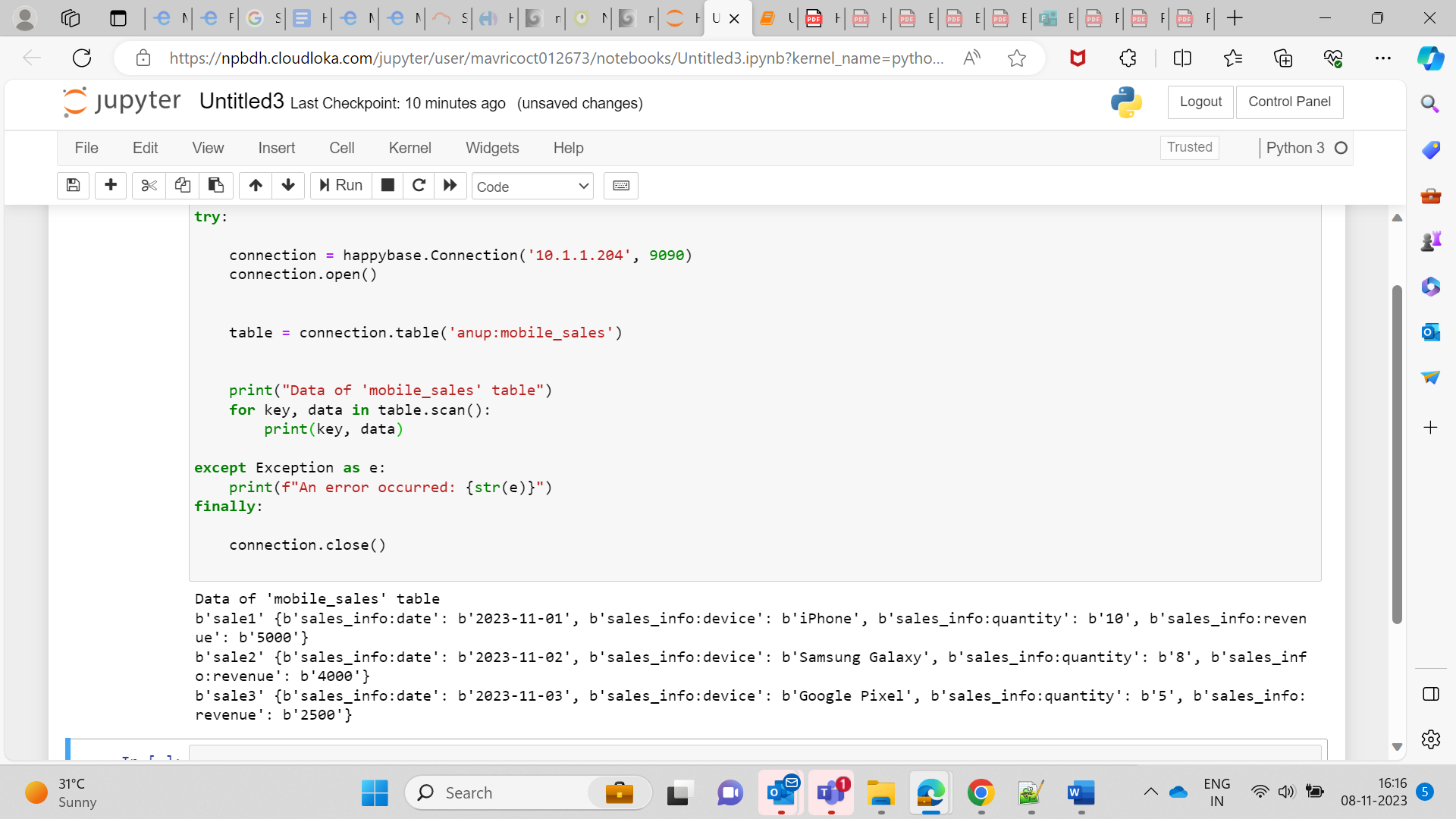
b.put(row\_key, data)

print("Inserted sample data into 'mobile\_sales' table")

except Exception as e:

print(f"An error occurred: {str(e)}")

finally:

connection.close()  
  
  
  
  
  
Retrieving all data from table  
  
  
  
import happybase

try:

connection = happybase.Connection('10.1.1.204', 9090)

connection.open()

table = connection.table('anup:mobile\_sales')

print("Data of 'mobile\_sales' table")

for key, data in table.scan():

print(key, data)

except Exception as e:

print(f"An error occurred: {str(e)}")

finally:

connection.close()