**Task1**

Access all the tables created in Assignment 1 and run those queries through the python API

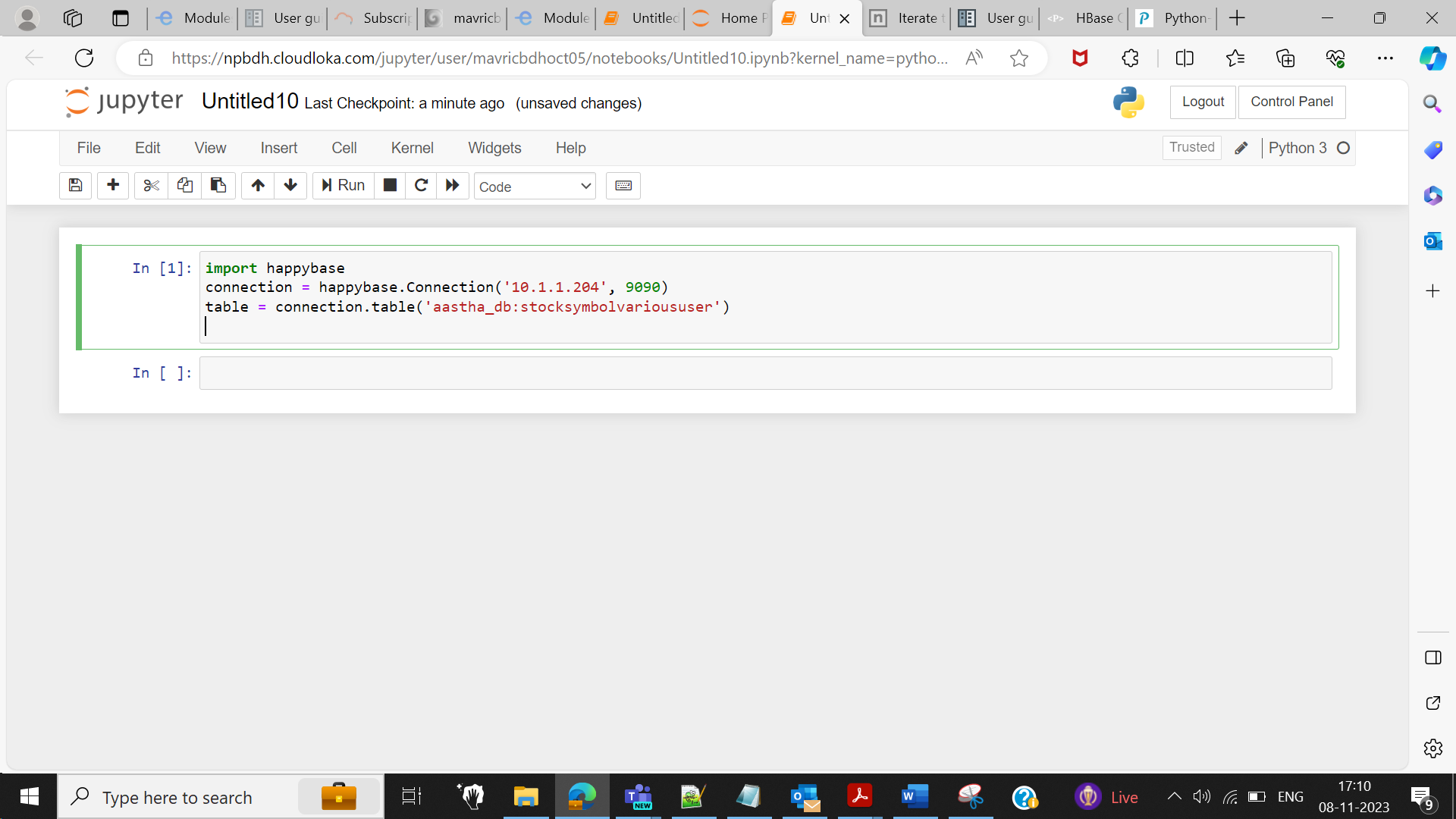
1. 'aastha\_db:stocksymbolvarioususer'
2. 'aastha\_db:album\_hbase

**Table1 results:**

import happybase

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

table = connection.table('aastha\_db:stocksymbolvarioususer')



for key, data in table.scan():

print(key, data)

b'1' {b'info:active': b'True', b'info:comment': b'I have this one on my radar.', b'info:commentid': b'comment1', b'info:displayname': b'KingJames', b'info:stock\_symbol': b'EXC', b'info:userid': b'user1'}

b'2' {b'info:active': b'True', b'info:comment': b'Interesting value proposition but too early to know if they will survive.', b'info:commentid': b'comment2', b'info:displayname': b'KingJames', b'info:stock\_symbol': b'EXC', b'info:userid': b'user1'}

b'3' {b'info:active': b'True', b'info:comment': b'like this one but would suggest waiting a few months.', b'info:commentid': b'comment1', b'info:displayname': b'Wizard of Wall Street', b'info:stock\_symbol': b'INFI', b'info:userid': b'user2'}

b'4' {b'info:active': b'False', b'info:comment': b'Not obvious yet but this stock is gonna be a huge winner.', b'info:commentid': b'comment2', b'info:displayname': b'Wizard of Wall Street', b'info:stock\_symbol': b'INFI', b'info:userid': b'user2'}

b'5' {b'info:active': b'False', b'info:comment': b'Buy it now before the rush.', b'info:commentid': b'comment1', b'info:displayname': b'Time World Series Winner', b'info:stock\_symbol': b'INFI', b'info:userid': b'user3'}

b'6' {b'info:active': b'True', b'info:comment': b'Dump this one fast!', b'info:commentid': b'comment2', b'info:displayname': b'Time World Series Winner', b'info:stock\_symbol': b'INFI', b'info:userid': b'user3'}

A screenshot of a computer

Description automatically generated

1. row = table.row(b'1', columns=[b'info:comment'])

print(row)

{b'info:comment': b'I have this one on my radar.'}

A screenshot of a computer

Description automatically generated

2.for key, data in table.scan(filter="SingleColumnValueFilter('info','userid',=,'substring:user3')"):

print(key,data)

b'5' {b'info:active': b'False', b'info:comment': b'Buy it now before the rush.', b'info:commentid': b'comment1', b'info:displayname': b'Time World Series Winner', b'info:stock\_symbol': b'INFI', b'info:userid': b'user3'}

b'6' {b'info:active': b'True', b'info:comment': b'Dump this one fast!', b'info:commentid': b'comment2', b'info:displayname': b'Time World Series Winner', b'info:stock\_symbol': b'INFI', b'info:userid': b'user3'}

A screenshot of a computer

Description automatically generated

**3. for key, data in table.scan(filter="SingleColumnValueFilter('info','active',=,'substring:False')"):**

**print(key,data)**

b'4' {b'info:active': b'False', b'info:comment': b'Not obvious yet but this stock is gonna be a huge winner.', b'info:commentid': b'comment2', b'info:displayname': b'Wizard of Wall Street', b'info:stock\_symbol': b'INFI', b'info:userid': b'user2'}

b'5' {b'info:active': b'False', b'info:comment': b'Buy it now before the rush.', b'info:commentid': b'comment1', b'info:displayname': b'Time World Series Winner', b'info:stock\_symbol': b'INFI', b'info:userid': b'user3'}

A screenshot of a computer

Description automatically generated

**Table2 results:**

import happybase

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

table = connection.table('aastha\_db:album\_hbase')

A screenshot of a computer

Description automatically generated

for key, data in table.scan():

print(key, data)

b'1' {b'info:duration': b'04:35', b'info:musicdirector': b'AK', b'info:singer': b'Singer1', b'info:song': b'Song1'}

b'2' {b'info:duration': b'03:54', b'info:musicdirector': b'JK', b'info:singer': b'Singer1', b'info:song': b'Song2'}

b'3' {b'info:duration': b'02:03', b'info:musicdirector': b'JK', b'info:singer': b'Singer1', b'info:song': b'Song3'}

b'4' {b'info:duration': b'01:54', b'info:musicdirector': b'AK', b'info:singer': b'Singer2', b'info:song': b'Song4'}

b'5' {b'info:duration': b'03:43', b'info:musicdirector': b'FG', b'info:singer': b'Singer2', b'info:song': b'Song5'}

b'6' {b'info:duration': b'02:03', b'info:musicdirector': b'JK', b'info:singer': b'Singer3', b'info:song': b'Song6'}

b'7' {b'info:duration': b'01:54', b'info:musicdirector': b'AK', b'info:singer': b'Singer3', b'info:song': b'Song1'}

b'8' {b'info:duration': b'03:43', b'info:musicdirector': b'FG', b'info:singer': b'Singer4', b'info:song': b'Song2'}

A screenshot of a computer

Description automatically generated

1.)for key, data in table.scan(filter="SingleColumnValueFilter('info','singer',=,'substring:singer2')"):

print(key,data)

b'4' {b'info:duration': b'01:54', b'info:musicdirector': b'AK', b'info:singer': b'Singer2', b'info:song': b'Song4'}

b'5' {b'info:duration': b'03:43', b'info:musicdirector': b'FG', b'info:singer': b'Singer2', b'info:song': b'Song5'}

A screenshot of a computer

Description automatically generated

**2.)for key, data in table.scan(filter="SingleColumnValueFilter('info','musicdirector',=,'substring:AK')"):**

**print(key,data)**

A screenshot of a computer

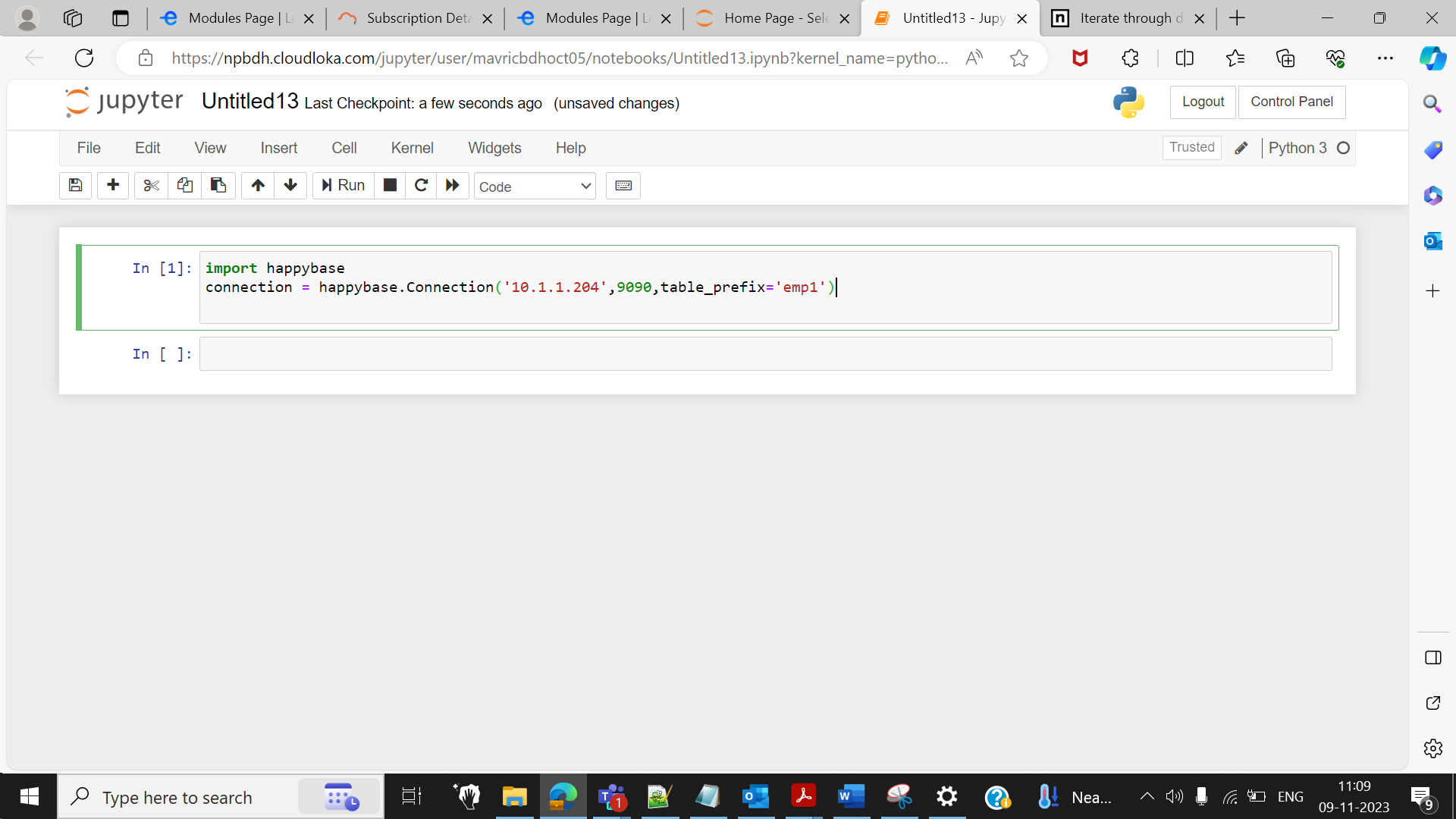
Description automatically generated

**TASK2**

Also Create new tables from python and insert some data in to those tables

**import happybase**

**connection = happybase.Connection('10.1.1.204',9090,table\_prefix='emp1')**



**connection.create\_table(**

**'employee12',**

**{'cf1': dict(max\_versions=10),**

**'cf2': dict(max\_versions=1, block\_cache\_enabled=False),**

**# use defaults**

**}**

**)**

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**table = connection.table('employee12')**

A screenshot of a computer

Description automatically generated

**for i in range(20):**

**table.put(b'rowkey-%d'%(i), {b'cf1:name': b'emp-%d'%(i),b'cf1:city': b'%d'%(i)})**

A screenshot of a computer

Description automatically generated

**for key, data in table.scan():**

**print(key, data)**

b'rowkey-0' {b'cf1:city': b'0', b'cf1:name': b'emp-0'}

b'rowkey-1' {b'cf1:city': b'1', b'cf1:name': b'emp-1'}

b'rowkey-10' {b'cf1:city': b'10', b'cf1:name': b'emp-10'}

b'rowkey-11' {b'cf1:city': b'11', b'cf1:name': b'emp-11'}

b'rowkey-12' {b'cf1:city': b'12', b'cf1:name': b'emp-12'}

b'rowkey-13' {b'cf1:city': b'13', b'cf1:name': b'emp-13'}

b'rowkey-14' {b'cf1:city': b'14', b'cf1:name': b'emp-14'}

b'rowkey-15' {b'cf1:city': b'15', b'cf1:name': b'emp-15'}

b'rowkey-16' {b'cf1:city': b'16', b'cf1:name': b'emp-16'}

b'rowkey-17' {b'cf1:city': b'17', b'cf1:name': b'emp-17'}

b'rowkey-18' {b'cf1:city': b'18', b'cf1:name': b'emp-18'}

b'rowkey-19' {b'cf1:city': b'19', b'cf1:name': b'emp-19'}

b'rowkey-2' {b'cf1:city': b'2', b'cf1:name': b'emp-2'}

b'rowkey-3' {b'cf1:city': b'3', b'cf1:name': b'emp-3'}

b'rowkey-4' {b'cf1:city': b'4', b'cf1:name': b'emp-4'}

b'rowkey-5' {b'cf1:city': b'5', b'cf1:name': b'emp-5'}

b'rowkey-6' {b'cf1:city': b'6', b'cf1:name': b'emp-6'}

b'rowkey-7' {b'cf1:city': b'7', b'cf1:name': b'emp-7'}

b'rowkey-8' {b'cf1:city': b'8', b'cf1:name': b'emp-8'}

b'rowkey-9' {b'cf1:city': b'9', b'cf1:name': b'emp-9'}

A screenshot of a computer

Description automatically generated