Name: Swetha Vijaya Raghavan

Last 4 digits ID: 1229

Computer IP address: 10.182.36.37

CSE 6331, Cloud Computing

Quiz Q7, Spring 2019 (c) DL, UTA, 2019

C&P means cut and paste only those relevant lines from your program(s) into this quiz.

1. I understand that I am on my honor during this quiz, I will not collaborate, use

non-allowed sources, and I will not discuss or share this quiz with anyone for the next

4 hours.

You MAY: use Google to search, use previous source code,

YOU MAY NOT use:

Email, Facebook, Hangouts, IM, chats, Skype or ANY other human connection.

This is a timed test. Late tests will have points deducted for being late.

Very late tests will not be graded.

When you are complete, with any part, please raise your hand, so we may visually inspect that part.

The second part of the test, you should electronically submit, you will need to copy and paste

only those lines of code to implement that part of the test, usually a few (two to eight) lines of code.

Place it immediately after the question.

Submit this Quiz (renamed) with code cut and pasted, ONLY text. DO NOT submit zips, binaries, libraries,

or anything other than text.

When any parts(questions) are complete complete, submit this test, you may make multiple submissions.

If you have computer, or other, problems, please raise your hand immediately.

If you understand and agree, please initial here:

\_\_\_\_\_\_\_\_svr\_\_\_\_\_\_\_\_\_

2. Total Points Possible:

If done on Google App Engine 100 Points, Total

If done on Google Cloud 75 Points, Total

If done on any other Cloud Service 40 Points, Total

If done locally (hosted locally) 10 Points, Total

(Or portion of total)

3. Name your program/submission with your name and last digits of your ID.

4. Using the cloud services provider, all functionality possible should be implemented on that provider.

(Of course, displaying web pages through a browser and the user interface is "local")

5. Show a web page and interface (which resides on the cloud provider) with your name and

student ID in large font at the top of every web page dispayed (for this quiz)

The cloud service provider is Google App Engine.

6. Import the pictures and data descriptions into the cloud provider. You may use any method and any data structure you wish.

Please note that for some parts of the following you will need to create indexes (keys) appropriately.

The data descrptions are a CSV file where the first entry is the picture (file) name, the second is the year created,

the third is the name of the creator, and the last are textual comments.

Show and submit code:

7. On a web page show all pictures with comments (from the metadata file) to the right of each picture, using buckets

Bucket Creation

def create\_bucket(bucket\_name):

"""Bucket Creation"""

storage\_client = storage.Client()

bucket = storage\_client.create\_bucket(bucket\_name)

print('Bucket {} created'.format(bucket.name))

def upload\_blob(bucket\_name, source\_file\_name, destination\_blob\_name):

storage\_client = storage.Client()

bucket = storage\_client.get\_bucket(bucket\_name)

blob = bucket.blob(destination\_blob\_name)

blob.upload\_from\_file(source\_file\_name)

@app.route('/q1')

def q1():

return render\_template('question1.html')

@app.route('/question1\_execute', methods=['POST'])

def question1\_execute():

file = request.files['file']

# upload\_blob('mybucket',file,'my\_image')

print(file.filename)

return 'success'

{% include 'common.html' %}

<form method="post" action="question1\_execute" enctype="multipart/form-data">

<input type="file" name="file"><br>

<input type="submit">

</form>

{% include 'common.html' %}

<form method="post" action="question2\_execute" enctype="multipart/form-data">

{# <input type="file" name="file"><br>#}

<input type="submit">

</form>

{% if result %}

{%for i in range(6)%}

<img src="./static/{{ url[i] }}" height="100" width="100">

<h3> {{year[i]}} </h3>

<h3> {{fname[i]}} </h3>

<h3> {{fdesc[i]}} </h3>

{%endfor%}

{% endif %}

@app.route('/q2')

def q2():

return render\_template('question2.html')

@app.route('/question2\_execute', methods=['POST'])

def question2\_execute():

sql = "SELECT \* FROM data"

cursor.execute(sql,)

result = cursor.fetchall()

print(result)

url = []

year = []

fname = []

fdesc = []

for row in result:

url.append(row['url'])

year.append(row['year'])

fname.append(row['name'])

fdesc.append(row['description'])

return render\_template('question2.html',result=result,url=url,year=year,fname=fname, fdesc=fdesc)





8. Allow a user (through a web form) to select creator (enter name) and year and show only those pictures

with picture name, year created and comments, using buckets.

@app.route('/q3')

def q3():

return render\_template('question3.html')

@app.route('/question3\_execute', methods=['GET'])

def question3\_execute():

year = str(request.args.get('year'))

cname = str(request.args.get('cname'))

if year !='' and cname !='':

print('1')

sql = "SELECT \* FROM data where name = " + "'" +cname + "'"+" and year = " + year

print(sql)

cursor.execute(sql)

result = cursor.fetchall()

lent = len(result)

url = []

year = []

fname = []

fdesc = []

for row in result:

url.append(row['url'])

year.append(row['year'])

fname.append(row['name'])

fdesc.append(row['description'])

# return render\_template('question3.html', result=result, url=url, year=year, fname=fname, fdesc=fdesc)

if year is not None and cname == '':

print('2')

sql = "SELECT \* FROM data where year = " + year

cursor.execute(sql)

result = cursor.fetchall()

lent = len(result)

url = []

year = []

fname = []

fdesc = []

for row in result:

url.append(row['url'])

year.append(row['year'])

fname.append(row['name'])

fdesc.append(row['description'])

if cname is not None and year == '':

print('2')

sql = "SELECT \* FROM data where name = " + "'" + cname + "'"

cursor.execute(sql)

result = cursor.fetchall()

lent = len(result)

url = []

year = []

fname = []

fdesc = []

for row in result:

url.append(row['url'])

year.append(row['year'])

fname.append(row['name'])

fdesc.append(row['description'])

return render\_template('question3.html', result=result, url=url, year=year, fname=fname, fdesc=fdesc,lent=lent)

{% include 'common.html' %}

<form method="get" action="question3\_execute" enctype="multipart/form-data">

{# <input type="file" name="file"><br>#}

year: <input type="text" name="year">

cname: <input type="text" name="cname">

<input type="submit">

</form>

{% if result %}

{%for i in range(lent)%}

<img src="./static/{{ url[i] }}" height="100" width="100">

<h3> {{year[i]}} </h3>

<h3> {{fname[i]}} </h3>

<h3> {{fdesc[i]}} </h3>

{%endfor%}

{% endif %}



9. Through a web interface allow individual comments to be modified, for a particular user name.

@app.route('/question9\_execute', methods=['GET'])

def question9\_execute():

description = str(request.args.get('description'))

cname = str(request.args.get('cname'))

if cname !='':

print('1')

sql = "update data set description = " + "'" +description + "'"+" where name = " + "'" +cname + "'"

print(sql)

cursor.execute(sql)

sql1 = "SELECT \* FROM data where name = " + "'" + cname + "'"

cursor.execute(sql1)

result = cursor.fetchall()

lent = len(result)

url = []

year = []

fname = []

fdesc = []

for row in result:

url.append(row['url'])

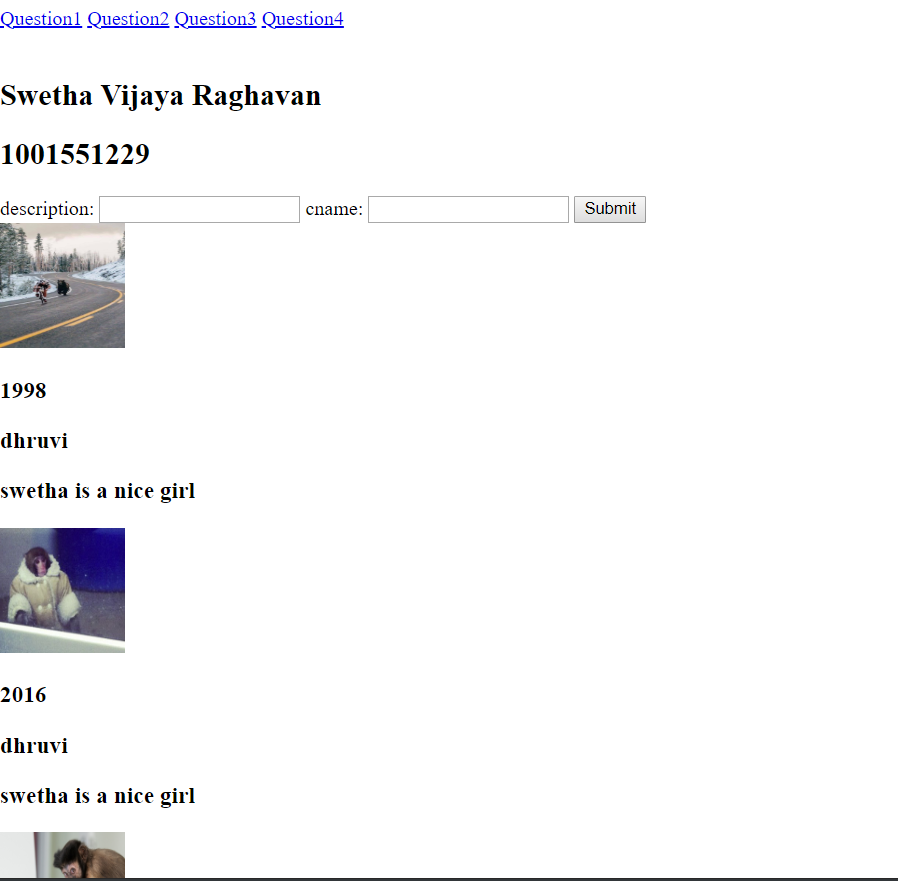
year.append(row['year'])

fname.append(row['name'])

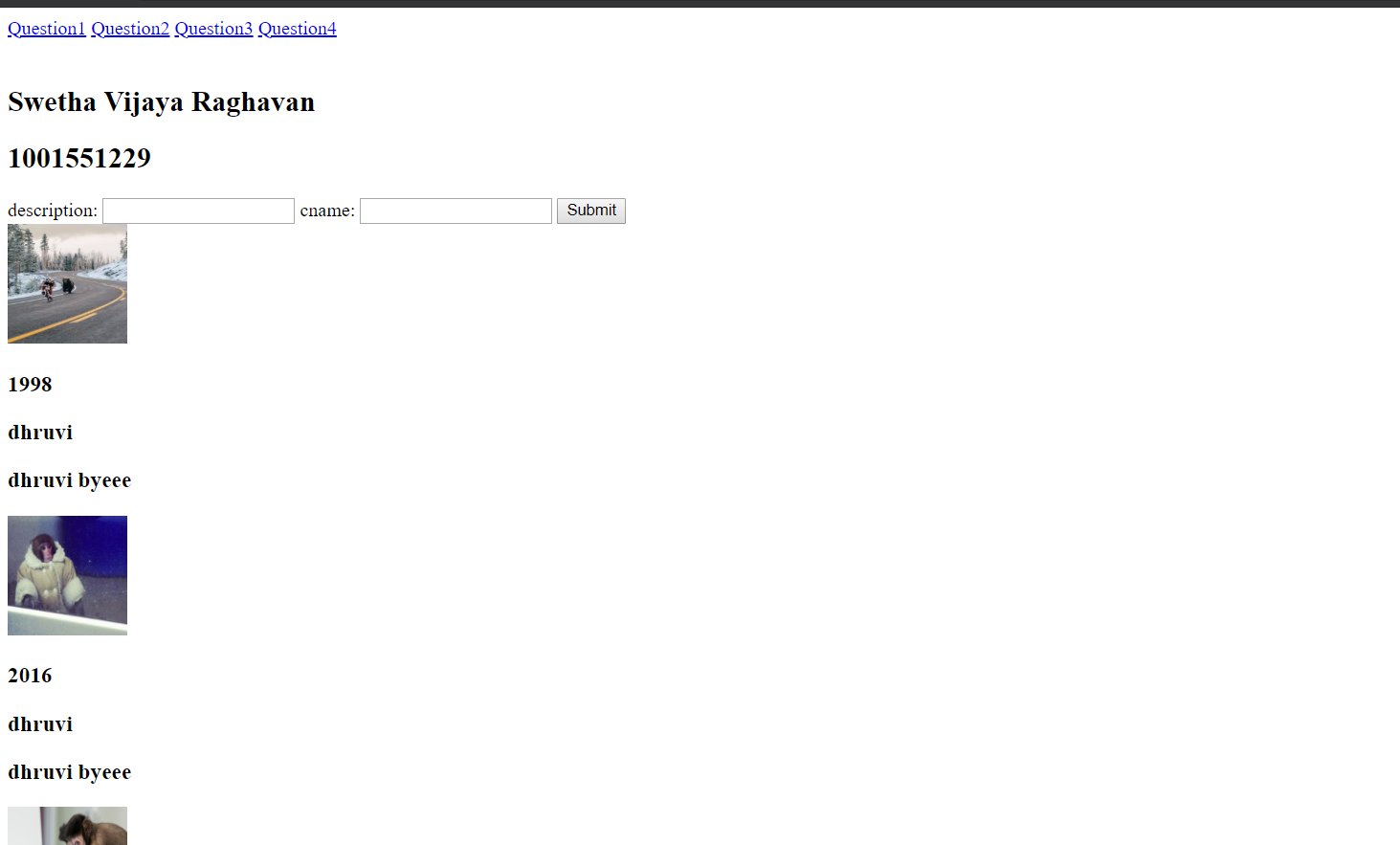
fdesc.append(row['description'])

return render\_template('question4.html', result=result, url=url, year=year, fname=fname, fdesc=fdesc,lent = len(result))

Before



After



10. Through a web form allow a user to give a "quota", then don't allow (or remove) files over this quota size (given in Kbytes).

11. Open two browsers, demonstrate the functionality in parts 8 and 9.

12. Show GTA parts (6 not necessary), 7, 8, 9, 10, 11

(as soon as you complete any part.)

13. When complete, return (send) this quiz

If you finish early, send this immediately, otherwise send between

the end of class and no more than 1 minute after that.

Good Luck!