\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
# v.01  
*import* sys  
*import* time  
*import* re  
*from* colorama *import* Fore  
*from* canvasapi *import* Canvas  
*import* urllib.request  
*import* csv  
*import* datetime  
  
  
API\_KEY = '7~eGmNWxkwKIWpZqedUamtX8vGGBMEqHDWnigiPDUi4xmvWjS3lZZJVO1hWMYK1dMO'  
API\_URL = "https://canvas.instructure.com"  
# initialize a new canvas object  
canvas = Canvas(API\_URL, API\_KEY)  
  
print(Fore.LIGHTBLUE\_EX + '\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*')  
print('\* MC Quiz Packer v.02 Creates MC Quizzes')  
print('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*'+Fore.RESET)  
  
*with* open('quizcontent.csv', newline='', encoding='utf8') *as* csvFile:  
 data = csv.reader(csvFile, dialect='excel', delimiter=',')  
 # creates a zip object of each 1st object in each row, etc.  
 columns = zip(\*data)  
 # creates a dictionary of those tuples  
 dataMap = {d[0]: d[1:] *for* d *in* columns}  
  
# Counts the quiz questions based on whether there is data in the question  
count = 0  
*for* listItem *in* dataMap.keys():  
 questionLookup = re.match(r'q\d{1,3}N', listItem)  
 *if* questionLookup:  
 listItem = dataMap.get(listItem)  
 # print(listItem[0])  
 *if* len(listItem[0]) > 0:  
 count = count + 1  
print(str('Creating ' + str(count) + ' MC Questions'))  
  
course\_id = ''.join(dataMap["course\_id"])  
title = ''.join(dataMap["title"])  
description = ''.join(dataMap["description"])  
time = ''.join(dataMap["time"])  
points = ''.join(dataMap["q\_Pts"])  
course = canvas.get\_course(course\_id)  
course\_name = course.name  
  
qt = datetime.datetime.now()  
qt = qt.strftime("%m%d%Y\_%H%M%S")  
uniqueQuizTitle = title + ' ' + qt  
  
print('For: ' + course\_name + ' (' + str(course\_id)+')')  
print('Quiz: ' + uniqueQuizTitle)  
print('Description: ' + description)  
print('Time to complete: ' + time)  
print('Points: ' + str(points))  
  
course.create\_quiz(quiz={'title': uniqueQuizTitle, 'description': description,  
 'time\_limit': time, 'shuffle\_answers': *True*,  
 'published': *True*, 'quiz\_type': 'assignment'})

######################  
# Retrieve the quiz id  
######################  
  
quizList = course.get\_quizzes()  
*for* quiz *in* quizList:  
 qTitle = quiz.title  
 *if* qTitle == uniqueQuizTitle:  
 quiz\_id = quiz.id  
 # print(quiz\_id)  
  
course = canvas.get\_course(course\_id)  
quiz = course.get\_quiz(quiz\_id)  
title = quiz.title  
  
###############################  
# question #  
###############################  
*while* count > 0:  
 count = str(count)  
 q1N = ''.join(dataMap["q"+count+"N"])  
 q\_Pts = ''.join(dataMap["q\_Pts"])  
 q1c1 = ''.join(dataMap["q"+count+"c1"])  
 q1c1\_w = ''.join(dataMap["q"+count+"c1\_w"])  
 q1c2 = ''.join(dataMap["q"+count+"c2"])  
 q1c2\_w = ''.join(dataMap["q"+count+"c2\_w"])  
 q1c3 = ''.join(dataMap["q"+count+"c3"])  
 q1c3\_w = ''.join(dataMap["q"+count+"c3\_w"])  
 q1c4 = ''.join(dataMap["q"+count+"c4"])  
 q1c4\_w = ''.join(dataMap["q"+count+"c4\_w"])  
  
 quiz.create\_question(question={'question\_text': q1N,  
 'question\_type': 'multiple\_choice\_question',  
 'points\_possible': q\_Pts,  
 'correct\_comments': 'Great Work!',  
 'incorrect\_comments': 'Im sorry thats not correct',  
 'neutral\_comments': 'Good Effort! You are on the right track.',  
 'text\_after\_answers': 'Thank you for answering this question!',  
 'answers[0][weight]': q1c1\_w,  
 'answers[0][answer\_text]': q1c1,  
 'answers[1][answer\_text]': q1c2,  
 'answers[1][weight]': q1c2\_w,  
 'answers[2][answer\_text]': q1c3,  
 'answers[2][weight]': q1c3\_w,  
 'answers[3][answer\_text]': q1c4,  
 'answers[3][weight]': q1c4\_w})  
 count = int(count)  
 count = count - 1