

Name: _____

CS3023 AY16 - Spring

Quiz 3

This is an open book exam. You may use any materials EXCEPT the Internet. You will be creating multiple files. Each question specifies the filename. Upload the files to the Quiz 3 assignment page in the course website.

Question 1

Consider the partially completed Pet class that has four attributes name, age, weight, and species. The incomplete source file [cs3023_quiz3_q1_pet.py](#) is available from the Quiz 3 Sakai page. You need to finish the class definition so the following test case (inside the file [pet_tester.py](#)) would pass. To save time, don't add anything that won't be tested in the following test cases. You don't need to worry about negative age or weight in the init method, i.e., assume the passed values are correct so you don't need to modify the given init method. Upload the completed file [cs3023_quiz3_q1_pet.py](#).

We are looking for a clean and efficient implementation. Points may be deducted for any cluttered or inefficient implementation.

```
def test_creation(self):
    p1 = Pet("Jack", 2, 3.5, Pet.DOG)
    p2 = Pet("Jill", 4, 2.3, Pet.CAT)
    self.assertTrue(p1.species != Pet.SNAKE)
    self.assertTrue(p2.species != Pet.RABBIT)

def test_walk(self):
    p1 = Pet("Jane", 2, 3.5, Pet.RABBIT)
    p2 = Pet("John", 5, 2.0, Pet.CAT)
    p1.eat()
    self.assertEqual(p1.weight, 3.6)
    p2.exercise()
    self.assertEqual(p2.weight, 1.8)

def test_str(self):
    p1 = Pet("Jack", 2, 3.0, Pet.DOG)
    p2 = Pet("Jill", 3, 2.5, Pet.CAT)
    p3 = Pet("Jane", 4, 3.0, Pet.SNAKE)
    p4 = Pet("John", 5, 2.5, Pet.RABBIT)
    self.assertEqual(str(p1), "Jack : 2 : 3.0 : Dog")
    self.assertEqual(str(p2), "Jill : 3 : 2.5 : Cat")
    self.assertEqual(str(p3), "Jane : 4 : 3.0 : Snake")
    self.assertEqual(str(p4), "John : 5 : 2.5 : Rabbit")
```

Question 2

Define a new class named **Kennel**. This class manages **Pet** objects. A kennel categorize pets into three separate groups based on their age. The first group includes pets with age between 0 and 5 inclusive, the second with those between 6 and 10 inclusive, and the third with those older than 10.

You define two methods:

1. **add** : This adds the passed Pet object to the correct group. Assume the argument is valid.

```
ken = Kennel()  
.  
.  
.  
ken.add(Pet("Jim", 2, 3.0, Pet.DOG))
```

2. **getGroup**: This returns a list of all those pets in a given group. The argument to this method is the group number 1, 2, or 3. Assume the argument is a valid number.

```
babies = ken.getGroup(1)
```

Download the starter file [cs3023_quiz3_q2_kennel.py](#). Complete the class and uploaded the file. You can complete this class without finishing Question 1.

We are looking for a clean and efficient implementation. Points may be deducted for any cluttered or inefficient implementation.

Question 3

- A. Define a class `Vehicle` with just the `__init__` method. The method accepts VIN (string) and owner name (string). The VIN must be a six-character string with the first three characters being uppercase alphabets and the last being three digits (e.g., "NAD983"). The owner name must be in the form of <last>, <first> (e.g., "Smith, Jack"). Just check that there's one comma in the string; don't worry about the first letter being uppercase or not. Implement the method so it raises (throws) a `ValueError` exception if there's any format violation in the passed values. Raise `TypeError` if any passed value is not a string.
- B. Write a code to create a new `Vehicle` object with an exception handling.

Save the file as `cs3023_quiz3_q3.py`. There's no starter file for this question.

Question 4

You should answer this question without actually running any code (although I won't prohibit you from doing so). Do not write your answer here. Submit your answer in the file [cs3023_quiz3_q4_answer.txt](#).

A. Consider the following code:

```
cnt = 0
room = Room()
hotel = [ ]
while cnt < 10:
    room.number = input("Room number:")
    room.occupant = input("Guest name:")
    hotel.append(room)
    cnt +=1
```

Assume the Room class is properly defined. What is the length of the list `hotel`, i.e., what is the value of `len(hotel)` after the while loop is executed? Explain. You need only a few sentences for your explanation.

B. Consider the following code:

```
r1 = Room()
r1.number = "G-107"
r1.occupant = "Mick Jaggar"
r2 = r1
r2.occupant = "Keith Richards"
assert r.occupant != r2.occupant
```

Does this assert statement raise an `AssertionError`? Yes or No. Explain your answer. Here also, you need only a few sentences to explain fully.