AllergicAdopter and MedicatedAllergicAdopter

(20 points possible)
The next two types of adopters will be the `AllergicAdopter` and the `MedicatedAllergicAdopter`.

The Allergic Adopter

The `AllergicAdopter` varies from the regular `Adopter` because an `AllergicAdopter` is extremely allergic to one or more particular species and cannot be around them even a little bit! If the adoption center contains one or more of those animals, they will not be able to go there. The `AllergicAdopter` is a **subclass of the `Adopter` class**, and should inherit from it and only it. The `AllergicAdopter`'s `__init__` method should look like the following:

```
__init__(self, name, desired_species, allergic_species)
```

All of the inputs are the same as the `Adopter` class, *except* that `allergic_species` is a **list of strings** of one or more species that the adopter is allergic to.

The `AllergicAdopter`'s scoring method also differs from the `Adopter`'s scoring method. You should override the method so that a score calculated on **an `AllergicAdopter` will return a value that is 0 if the adoption center has one or more of a species that the adopter is allergic to, otherwise it should calculate score based on the `Adopter`'s calculate score method.** Note that since `allergic_species` is a list, you will have to iterate over the values to check if the `AdoptionCenter` contains one or more of any. The scoring method should take only one argument, the `AdoptionCenter` instance to calculate the score from.

Below, please write your implementation of the `AllergicAdopter` class, including its `__init__` method and its `get_score(adoption_center)` method.

The Medicated Allergic Adopter

The `MedicatedAllergicAdopter` varies from the `AllergicAdopter` as they have medicine to lessen their allergies. The `MedicatedAllergicAdopter` is a **subc of the `AllergicAdopter` class**, and should inherit from the `MedicatedAllergicAdopter`'s `__init__` method should look like the following:

```
__init__(self, name, desired_species, allergic_species, medicine_effectiveness)
```

All of the inputs are the same as the `AllergicAdopter` class, *except* that medicine_effectiveness is a **dictionary of {string: float}** of the medicines effectiveness to certain species. The effectiveness can range from 0.0 (no effectiveness against allergies) to 1.0 (full effectiveness against allergies).

For example, medicine_effectiveness may look like `{"Dog": 0.5, "Cat": 0.0, "Horse": 1.0}`, which means there is a medium effectiveness against dog allergies, no effectiveness against cat allergies, and full effectiveness against horse allergies.

The `MedicatedAllergicAdopter`'s scoring method also differs from the `AllergicAdopter`'s scoring method. Since the `MedicatedAllergicAdopter` is able to prevent against some allergies, they are now able to enter some `AdoptionCenters` they could not before. To calculate the score for a specific adoption center, we want to find what is the most allergy-inducing species that the adoption center has for the particular `MedicatedAllergicAdopter`. To do this, first examine what species the AdoptionCenter has that the `MedicatedAllergicAdopter` is allergic to, then compare them to the `medicine_effectiveness` dictionary. Take the lowest `medicine_effectiveness` found for these species, and multiply that value by the `Adopter`'s calculate score method.

For example, consider the following:

Joe is allergic to dogs and horses, but wants a cat. He takes a medicine that has 0.5 effectiveness against dog allergies, and 1.0 effectiveness against horse allergies. He is considering going to an adoption center that has dogs, cats, and horses. Since the adoption center contains both of his allergies, to calculate his score, we will want to take the lowest effectiveness, that is, the 0.5 effectiveness against dogs, and multiply it by the normal Adopter score. The end score for his would be 0.5 * the base class `Adopter` score.

Below, please write your implementation of the `MedicatedAllergicAdopter` class, including its `__init__` method and its `get_score(adoption_center)` method.

```
1 # Enter your code for the AllergicAdopter and MedicatedAllergicAdopter classes here
2
```

Unanswered

You have used 0 of 30 submissions