



Bookmarks



Bookmark

► Overview

► Entrance
Survey

► Week 1

► Week 2

► Week 3

► Week 4

► Quiz

► Week 5

► Week 6

▼ Week 7

Lecture 13 - Trees -

Time 51:54

Lecture Sequence

Wrap up - Time

33:39

Problem Set 7

Problem Set due Aug
04, 2016 at 23:30 UTC

► Sandbox

Week 7 > Problem Set 7 > Part 2A - The Adopter

Meet the Adopter

(10 points possible)

There are a few types of potential adopters. The base class of the adopters is simply called "Adopter", which you will write below. The Adopter class will contain information that will be shared among all types of adopters.

Adopter Initialization

The following information should be stored in an `Adopter` instance, and passed in as its initialization variables:

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

- `name` - A string that represents the name of the adopter
- `desired_species` - A string that represents the desired species to adopt

Adopter Methods

The following methods should be implemented for the `Adopter` class

- `get_name()` - Returns the name of the adopter
- `get_desired_species()` - Returns the desired species of the adopter
- `get_score(adoption_center)` - Returns the score (details below)

About Scoring

Each `Adopter` class, and each `Adopter` subclass will have its own scoring methods. The minimum value that a score can be is 0, and there is no upper bound. The score method will take in an `adoption_center` as its argument, and will do some calculations to determine how good of a fit the specific adopter is to the specific adoption center.

For the base `Adopter` class, this score will be $1 * \text{num_desired}$ where *num_desired* is the number of the adopter's desired species that the adoption center has.

```
1 # Enter your code for the Adopter class here
2
```



Unanswered

You have used 0 of 30 submissions

© All Rights Reserved



© edX Inc. All rights reserved except where noted. EdX, Open edX and the edX and Open EdX logos are registered trademarks or trademarks of edX Inc.

POWERED BY
OPENedX

