



Day 4: Class vs. Instance ☆

2 more challenges to get your next star!

Points: 5/7

30
Days of Code

Problem

Submissions

Leaderboard

Editorial

Tutorial

RATE THIS CHALLENGE



Objective

In this challenge, we're going to learn about the difference between a class and an instance; because this is an Object Oriented concept, it's only enabled in certain languages. Check out the [Tutorial](#) tab for learning materials and an instructional video!

Task

Write a Person class with an instance variable, **age**, and a constructor that takes an integer, **initialAge**, as a parameter. The constructor must assign **initialAge** to **age** after confirming the argument passed as **initialAge** is not negative; if a negative argument is passed as **initialAge**, the constructor should set **age** to 0 and print `Age is not valid, setting age to 0.` In addition, you must write the following instance methods:

1. `yearPasses()` should increase the **age** instance variable by 1.
2. `amIOld()` should perform the following conditional actions:
 - If **age** < 13, print `You are young.`
 - If **age** ≥ 13 and **age** < 18, print `You are a teenager.`
 - Otherwise, print `You are old.`

To help you learn by example and complete this challenge, much of the code is provided for you, but you'll be writing everything in the future. The code that creates each instance of your Person class is in the main method. Don't worry if you don't understand it all quite yet!

Note: Do not remove or alter the stub code in the editor.

Input Format

Input is handled for you by the stub code in the editor.

The first line contains an integer, **T** (the number of test cases), and the **T** subsequent lines each contain an integer denoting the **age** of a Person instance.

Constraints

- $1 \leq T \leq 4$
- $-5 \leq \text{age} \leq 30$

Output Format

Complete the method definitions provided in the editor so they meet the specifications outlined above; the code to test your work is already in the editor. If your methods are implemented correctly, each test case will print 2 or 3 lines (depending on whether or not a valid **initialAge** was passed to the constructor).

Sample Input

```
4
-1
10
16
18
```

Sample Output

```
Age is not valid, setting age to 0.
You are young.
You are young.

You are young.
You are a teenager.

You are a teenager.
You are old.

You are old.
```



You are old.

Explanation

Test Case 0: **initialAge = -1**

Because **initialAge < 0**, our code must set **age** to **0** and print the "Age is not valid..." message followed by the young message. Three years pass and **age = 3**, so we print the young message again.

Test Case 1: **initialAge = 10**

Because **initialAge < 13**, our code should print that the person is young. Three years pass and **age = 13**, so we print that the person is now a teenager.

Test Case 2: **initialAge = 16**

Because **13 ≤ initialAge < 18**, our code should print that the person is a teenager. Three years pass and **age = 19**, so we print that the person is old.

Test Case 3: **initialAge = 18**

Because **initialAge ≥ 18**, our code should print that the person is old. Three years pass and the person is still old at **age = 21**, so we print the old message again.

The extra line at the end of the output is supposed to be there and is trimmed before being compared against the test case's expected output. If you're failing this challenge, check your logic and review your print statements for spelling errors.

Change Theme

Python 3



```
1 class Person:
2     age=0
3     def __init__(self,initialAge):
4         # Add some more code to run some checks on initialAge
5         if initialAge < 0:
6             print("Age is not valid, setting age to 0.")
7         else:
8             self.age = initialAge
9     def amIOld(self):
10        # Do some computations in here and print out the correct statement to the console
11        if self.age < 13:
12            print("You are young.")
13        elif self.age >= 13 and self.age < 18:
14            print("You are a teenager.")
15        else:
16            print("You are old.")
17    def yearPasses(self):
18        # Increment the age of the person in here
19        self.age += 1
20
21 > t = int(input()) ...
```

Line: 1 Col: 1

Upload Code as File ☐ Test against custom input

Run Code

Submit Code

Congratulations!

You have passed the sample test cases. Click the submit button to run your code against all the test cases.

✓ Sample Test case 0

Your Output (stdout)

```
1 Age is not valid, setting age to 0.
2 You are young.
3 You are young.
4
5 You are young.
6 You are a teenager.
7
```



8 You are a teenager.
9 You are old.
10
11 You are old.
12 You are old.
13

