Leaderboard

Points:

Discussions



Submissions

Day 4: Class vs. Instance 🌣

3 more challenges to get your next star!

Editorial 🖰

Tutorial

		30
4/7	1	Days of Code ★

Objective

Problem

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 ageafter confirming the argument passed as initial Age is not negative; if a negative argument is passed as initial Age, 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 $oldsymbol{1}$.
- 2. amIOld() should perform the following conditional actions:
 - \circ If age < 13, print You are young..
 - \circ If $age \geq 13$ 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.

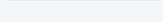
The first line contains an integer, T (the number of test cases), and the T subsequent lines each contain an integer denoting the age of a Person instance.

Constraints

- $1 \le T \le 4$
- $-5 \le age \le 30$

Output Format

Author	blondiebytes
Difficulty	Easy
Max Score	30
Submitted By	229633



NEED HELP?

- View tutorial
- View discussions
- View editorial
- View top submissions

RATE THIS CHALLENGE



MORE DETAILS

- Download problem statement
- Download sample test cases
- **Suggest Edits**





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 ${\bf 2}$ or ${\bf 3}$ lines (depending on whether or not a valid initial Age 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 $13 \leq initial Age < 18$, 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 $initialAge \geq 18$, 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 Java 8 1 ⊞ import java.io.*; ··· 3 public class Person { 4 private int age; 5 public Person(int initialAge) { // Add some more code to run some checks on initialAge } 9 10 public void amIOld() { 11 // Write code determining if this person's age is old and print the correct 12 statement: System.out.println(/*Insert correct print statement here*/); 13 } 14 15 public void yearPasses() { 16 // Increment this person's age. 17 } 18 19 public static void main(String[] args) { 20 E Scanner sc = new Scanner(System.in); 21 int T = sc.nextInt(); 22 23 for (int i = 0; i < T; i++) { int age = sc.nextInt(); 24 Person p = new Person(age); 25 p.amIOld(); 26 for (int j = 0; j < 3; j++) { 27 p.yearPasses(); 28 29 p.amIOld(); System.out.println(); 31 32 sc.close(); 33 34 35

1 Upload Code as File

☐ Test against custom input

Run Code

Submit Code

Contest Calendar | Blog | Scoring | Environment | FAQ | About Us | Support | Careers | Terms Of Service | Privacy Policy | Request a Feature