

Name: Akash Singh
Employee Code: 1874

HackerRank: 30 Days of Code
Day 4: Class vs. Instances

Day 4: Class vs. Instances:

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:
 - a) If age < 13, print You are young.
 - b) If age >= 13 and age < 18, print You are a teenager.
 - c) 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 !

Solution:

```
import java.io.*;
import java.util.*;

public class Person {
    private int age;

    public Person(int initialAge)
    {
        // Add some more code to run some checks on initialAge
        if (initialAge > 0)
        {
            age = initialAge;
        }
        else
        {
            System.out.println("Age is not valid, setting age to 0.");
        }
    }
}
```

```

        age = 0;
    }
}

public void amIOld() {
    // Write code determining if this person's age is old and
    print the correct statement:
    if (age < 13){
        System.out.println("You are young.");
    }
    else if (age < 18){
        System.out.println("You are a teenager.");
    }
    else{
        System.out.println("You are old.");
    }
}

public void yearPasses() {
    // Increment this person's age.
    age++;
}

public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
    int T = sc.nextInt();
    for (int i = 0; i < T; i++) {
        int age = sc.nextInt();
        Person p = new Person(age);
        p.amIOld();
        for (int j = 0; j < 3; j++) {
            p.yearPasses();
        }
        p.amIOld();
        System.out.println();
    }
    sc.close();
}
}

```

Output:

Test case 0 Compiler Message

Success

Input (stdin)

```
1 4
2 -1
3 10
4 16
5 18
```

Test case 0 Expected Output

```
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.
```

HackerRank Prepare > Tutorials > 30 Days of Code > Day 4: Class vs. Instance

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:

- `yearPasses()` should increase the `age` instance variable by 1.
- `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

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

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.

30 You have earned 30.00 points!
You are now 2 challenges away from the 2nd star for your 30 days of code badge.

Congratulations

You solved this challenge. Would you like to challenge your friends? [Facebook](#) [Twitter](#) [LinkedIn](#)

The next challenge in this tutorial will unlock in 07:32:17

[Go to Dashboard](#) [Try a Random Challenge](#)

Test case 0 Compiler Message

Success

Input (stdin)

```
1 4
2 -1
3 10
4 16
5 18
```

Expected Output

```
1 Age is not valid, setting age to 0.
```

Result:

Program executed successfully .