

Programming Assignment

In your `case1`, `case2` and `case3` packages, you will find two classes, `Person` and `PersonWithJob`. The code for these classes is reproduced below:

You will modify these classes by **overriding** the **`equals`** method in three different ways.

Case 1: For the code in package `case1`, override `equals` using the *instanceof strategy*. Two `Persons` are considered equal if they have the same name and date of birth, and this definition for `equals` will be considered in this case to be an adequate criterion for deciding if two `PersonWithJob` instances are equal; that is, two `PersonWithJob` instances are also considered equal if they have the same name and date of birth.

Case 2: For the code in package `case2`, `Person` and `PersonWithJob` must be equipped with different `equals` methods. As in Case 1, two `Persons` should be considered equal if they have the same name and date of birth. However, in Case 2, two instances of `PersonWithJob` will be considered equal only if they have the same name, date of birth, and *salary*. Implement this requirement using the *same classes strategy*.

Case 3: For the code in package `case3`, `Person` and `PersonWithJob` must be equipped with different `equals` methods, exactly as in Case 2. However, in this case, you must handle this requirement by using **composition** instead of inheritance.