

 <p>University of Windsor</p> <p>SCHOOL OF COMPUTER SCIENCE</p>	<p>COMP-2120, Fall 2024</p> <p>Object Oriented Programming</p> <p>Using Java</p> <p>Assignment 2</p> <p>TOTAL MARKS: 32</p>
--------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------

LAND ACKNOWLEDGEMENT

The School of Computer Science at the University of Windsor *sits on the Traditional Territory of the Three Fires Confederacy of First Nations*. We acknowledge that this is the beginning of our journey to understanding the Significance of the history of the Peoples of the Ojibway, the Odawa, and the Pottawatomie.

SUBMISSION DEADLINE: 11:59 PM, NOV-29, 2024

NOTE: If you need additional time, you can complete your submission by Dec 5 without any penalty.

SUBMISSION GUIDELINE

1. This is an individual assignment. You are allowed to discuss the assignment but you cannot copy code from others/ share code with others. Any form of plagiarism will lead to a score of 0 for your assignment.
2. Ensure that your source code is properly formatted (i.e., code has proper indentation, whitespaces, and meaningful variable names).
3. Add comments to your code to explain what the code is doing.
4. Include your first name, last name and student number at the beginning of each file using comment. You can also use Javadoc comments (@author) to include the information.

PROBLEM

In the previous assignment, you created a program that allow users (i.e., players) to guess the birthday of a person (e.g., Terry Fox was born on July 28, 1958). In this assignment, you need to modify your program and prepare another version of the program. Please carefully check the following instructions.

IMPLEMENTATION DETAILS

1. Modify the Person class to be an abstract class with four instance variables: String name, Date birthday, and String country, int difficulty. The value of the difficulty variable can be one of the following three values: 1, 2, 3. **(1 Mark)**
2. Create a constructor with one argument for the Person class that initializes the difficulty variable. **(1 Mark)**
3. Add an abstract method personType. The method takes no parameter and returns a String. It will be implemented in the derived concrete classes to return a short description of the entity type, as described later in this assignment. **(1 Mark)**
4. Add another abstract method called clone. This method takes no parameter but returns an object of type Person. **(1 Mark)**

5. Add a method called `welcomeMessage` that takes no parameter but returns a string of the following format: `Welcome! Guess the birthday of X named Y.` Here, X will be replaced by the String returned by the `personType` method (you need to call this method) and Y will be replaced by the name of the person. Call this method at the beginning of guessing the birthday of a person. **(2 Marks)**
6. Add a method, called `successMessage` that takes no parameter but returns a string of the following format: `Wow! You are able to successfully guess the birthday of X` where X will be replaced with the String obtained by calling the `toString` method of the class. Call this method when a player is able to guess the birthday correctly. **(2 Marks)**
7. Create two classes: `Politician` and `Player` both are subclasses of the `Person`.
8. The `Politician` class has one attribute, called `party`, which is of type `String` and holds the name of the political party of the politician. **(1 Mark)**
9. The `Player` class has two attributes. The first one is of type `String` (called `game`) that holds the name of the game. The second one is also of type `String` that holds the name of the team (called `team`). **(1 Mark)**
10. Define a `toString()` method in the `Politician` class that will create a string from the string obtained by calling the `toString` method in the parent class and append that with the following string: `X is a member of Party Y.` Here, X is the name of the politician and Y is the name of the party. **(2 Marks)**
11. Define a `toString()` method in the `Player` class that will create a string from the string obtained by calling the `toString` method in the parent class and append that with the following string: `X is a Player of Team Y.` Here, X is the name of the player and Y is the name of the team. **(2 Marks)**
12. Implement proper constructors, including a copy constructor, for both subclasses that can initialize instance variables in `Player` and `Politician` including those inherited by its parent class. For determining the order of the parameters refer to the test code provided at the end of the assignment. To do so, you need to use a super constructor. The copy constructors will need to invoke the copy construct in its parent class. **(2 +2 = 4 Marks)**
13. Implement the `personType` method in the `Player` and `Politician` subclasses. The `personType` method in the `Player` class will return the string *player* and the `PersonType` method in the `Politician` class will return the string *politician*. **(2 Marks)**
14. Implement the `clone` method in the `Player` and `Politician` subclasses **(2 Marks)**
15. Now, create a class, called `GuessMaker2` that will control the game. The class `GuessMaker2` should contain an `ArrayList` to maintain a list of `Persons` to guess from (instead of an array you used earlier). `GuessMaker2` should also have an `int` variable, called `currentPerson` that points to the index of the `Person` object in the array that is used to play the game. The value of the variable needs to be updated to reflect any changes in the game (i.e., player is guessing the birthday of another person). **(2 Marks)**
16. `GuessMaker2` should have a public method called `addPerson` that returns nothing but an `Object` of type `Person` as a parameter and add that to the `ArrayList` of persons. **(1 Mark)**
17. Create a method: `public void startGame()` that controls the game loop. Ensure that the loop is working correctly.
18. Use the `difficulty` variable as follows: If the `difficulty` level is set to 3, the program will guide the player whether they need to select an earlier or later year, month and date by displaying the appropriate message (e.g., select an earlier year, select a later month, select

a later date). In other words, the program provides suggestions for year, month and date. If the difficulty level is set to 2, the program assists in selecting year and month only. If the difficulty level is set to 1, the program assists in selecting the year only. **(3 Marks)**

19. We will insert the following code in the main method of the GuessMaker2 to test your implementation:

```
Politician trudeau = new Politician("Justin Trudeau", new Date(25, 12, 1971), 1,
"Liberal");
```

```
Player ronaldo = new Player("Leo Messi", new Date(24, 6, 1986), 2,
"Soccer", "Barcelona");
```

```
Politician pierre = new Politician("Pierre Poilievre", new Date(3, 6, 1979), 3,
"Conservative");
```

```
GuessMaker2 gm = new GuessMaker2();
```

```
gm.addPerson(trudeau);
```

```
gm.addPerson(ronaldo);
```

```
gm.addPerson(pierre);
```

```
gm.startGame();
```

You will obtain **2 Marks** if the above test code runs correctly and can generate the correct output. **(2 Marks)**

20. Add any other methods you need to complete your implementation or reuse any code you implemented earlier. However, make sure you follow the instructions in this assignment
21. Your program should be able to handle invalid input.

DISTRIBUTION OF MARKS

1. Complete the implementation of Person, Player and Politician classes as described above **(22 Marks)**
2. Implement the GuessMaker2 correctly with the necessary changes **(8 Marks)**
3. Format and comment your code (including adding author information) **(2 Marks)**

WHAT DO YOU NEED TO DO?

1. Complete the program.
2. Put all the files inside a single folder.
3. Rename the folder as follows: FirstName_LastName_StudentNumber_Assignment2
4. Zip up your folder and submit it under the "Assignment 2" submission on Brightspace.