

1.

Untuk mendaftarkan warga kelurahan, anda diminta untuk membuat kelas-kelas yang dapat menjelaskan nasionalitas seseorang. Kewarganegaraan seseorang dimodelkan sebagai kelas `Citizenship` yang merupakan kelas abstrak. Ada 3 tipe warga negara, yaitu `Indonesian`, `American`, dan `Japanese`. Warga negara Indonesia itu sendiri memiliki bahasa daerah yang berbeda-beda, maka anda diminta untuk menggunakan `IndonesianLocalLanguage.java` untuk membedakan bahasa daerah yang dimiliki warga setempat.

Dengan memanfaatkan *template* yang dilampirkan pada akhir soal, anda diminta untuk mengimplementasikan kelas `Citizenship.java` yang merupakan abstract class. Kemudian implementasikan juga `Indonesian.java`, `Japanese.java`, dan `American.java` yang merupakan turunan dari `Citizenship.java`.

Kelas `Citizenship` memiliki atribut sebagai berikut:

1. `surname`, yang berarti nama keluarga orang tersebut
2. `givenName`, yang berarti nama pertama orang tersebut

Kelas `Citizenship` juga memiliki metode sebagai berikut:

1. `getSurname`, yang mengembalikan `surname` citizen
2. `getGivenName`, yang mengembalikan `givenName` citizen
3. `speak`, yang merupakan metode **abstrak** dan tidak mengembalikan apapun.

Kelas `American` memiliki atribut `state` yang menandakan daerah / state yang ditinggali oleh orang tersebut. Kelas `American` juga memiliki metode sebagai berikut:

1. `getState`, yang mengembalikan `state` dari `American`
2. `speak`, yang mencetak "Hello buddy!" diakhiri newline.

Kelas `Indonesian` memiliki atribut `localLanguage` dengan tipe `IndonesianLocalLanguage`. Selain itu, kelas ini juga memiliki metode sebagai berikut:

1. `getLocalLanguage`, yang mengembalikan `localLanguage` dari `Indonesian`,
2. `speakLocalLanguage`, yang mencetak ke layar "Nami Kulo Tuan Mal" bila `localLanguage` bertipe **JAVANESE**, dan "Nepangkeun wasta sim kuring Tuan Mal" bila `localLanguage` bertipe **SUNDANESE**. Keduanya diakhiri oleh newline
3. `speak`, yang mencetak "Halo, aku Tuan Mal" diakhiri newline.

Kelas `Japanese` memiliki atribut `favouriteAnime` dengan tipe `string`. Selain itu, kelas ini juga memiliki metode sebagai berikut:

1. `getFavouriteAnime`, yang mengembalikan anime favorit dari `Japanese`,
2. `speak`, yang mencetak "Hai, watashi wa Mal-san desu!" diakhiri newline.

Implementasikan semua metode yang dijelaskan diatas dengan memanfaatkan *template* yang diberikan dan zip semua file (`Citizenship.java`, `American.java`, `Indonesian.java`, dan `Japanese.java`) dalam file `Citizenship.zip`.

Submit file `Citizenship.zip`

Attachments: `attachments.zip`



`Citizenship.zip`

<https://drive.google.com/file/d/1M0BN8AVEeX08PXdiZSIDc948QSVzfdOW/view?usp=sharing>

2.

Buatlah kelas `CitizenPrinter` yang berguna untuk mencetak informasi `Citizenship` dan kelas turunannya (`American`, `Japanese`, `Indonesian`) dari beberapa warga. Menggunakan kelas yang telah dibuat sebelumnya, kelas `CitizenPrinter` memiliki atribut `citizenshipList` yang berbentuk `ArrayList of Citizenship`. `CitizenPrinter` juga memiliki metode sebagai berikut:

1. `addCitizen`, yang menambahkan satu citizen ke dalam atribut `citizenshipList` dan tidak mengembalikan apapun
2. `getCitizenshipList`, yang mengembalikan atribut `citizenshipList`
3. `printCitizenshipList`, yang akan melakukan iterasi terhadap isi `citizenshipList`, kemudian, untuk setiap `citizenship`, dia akan secara berturut-turut melakukan hal berikut:
 1. Tergantung *instance* dari `Citizenship`:
 - Untuk kelas `American`, cetak "American" diakhiri new line
 - Untuk kelas `Indonesian`, cetak "Indonesian" diakhiri new line
 - Untuk kelas `Japanese`, cetak "Japanese" diakhiri new line
 2. Mencetak nama yang dipisahkan spasi, contohnya misal `surname` = "Tuan" dan `givenName` = "Mal" maka akan mencetak "Tuan Mal" diakhiri new line
3. Memanggil metode `speak`
4. Tergantung *instance* dari `Citizenship`:
 - Untuk kelas `American`, cetak nilai atribut `state` diakhiri new line
 - Untuk kelas `Indonesian`, panggil metode `speakLocalLanguage`
 - Untuk kelas `Japanese`, cetak nilai atribut `favouriteAnime` diakhiri new line

Lengkapi file `CitizenPrinter.java`

Submit file `CitizenPrinter.java`

/**

```

* CitizenPrinter.java
* [jelaskan kegunaan file ini]
* @author [NIM] [Nama]
*/
import java.util.ArrayList;
import java.util.List;

public class CitizenPrinter {
    private List<Citizenship> citizenshipList = new ArrayList<Citizenship>();

    /**
     * Add Citizen. Menambahkan citizenship ke array
     *
     * @param citizenship objek Citizenship
     */
    public void addCitizen(Citizenship citizenship) {
        // Tambahkan citizenship ke list, gunakan metode add dari java.util.List
    }

    /**
     * Get Citizenship. Getter dari citizenshipList
     */
    public List<Citizenship> getCitizenshipList() {
        // Lengkapi getter
    }

    /**
     * Print Citizenship List. Cetak informasi dan lakukan aksi dari citizenship yang
     ada.
     */
    public void printCitizenshipList() {
        for (Citizenship citizenship : this.citizenshipList) {
            // Lengkapi Print Citizenship List, hint:
            // - gunakan instanceof untuk mengetahui tipe dari objek
            // - gunakan type casting untuk memanggil metode kelas turunan
        }
    }
}

```

Time limit	1 s
Memory limit	64 MB

Diberikan *interface* sebagai berikut:

1. *Computer* (*Computer.java*)
2. *Monitor* (*Monitor.java*)
3. *Keyboard* (*Keyboard.java*)
4. *Portable* (*Portable.java*)

Diberikan juga *class* yang merepresentasikan jenis komputer sebagai berikut:

1. *GameConsole* (*GameConsole.java*) yang mengimplementasikan *Computer* dan *Monitor*.
2. *Laptop* (*Laptop.java*) yang mengimplementasikan *Computer*, *Keyboard*, dan *Portable*.
3. *PC* (*PC.java*) yang mengimplementasikan *Computer*, *Monitor*, dan *Keyboard*.
4. *Smartphone* (*Smartphone.java*) yang mengimplementasikan *Computer* dan *Portable*.

Lengkapilah metode-metode pada file *ComputerManager.java*.

Submit file *computerManager.java*

Attachments: *attachments.zip*

<https://drive.google.com/file/d/13ZI4b7Oej40zxSRbjaBFFzS-3U0cS4/view?usp=sharing>