SE116 – LAB#7 2023-2024 SPRING

Aim: Understanding the use of interface.

1. In order to simulate a smartphone OS system, implement the design stated below:

Write an interface `Playable` with the following abstract methods: `void play(), void stop(), void next()` and `void prev()`.

Write a class `Radio` that `implements Playable`. The class `Radio` represents a radio app to be used in our smartphone. The class `Radio` has the following private data members: `String channelName, String programName` and `int volumeLevel`. The class `Radio` defines all the abstract methods of `Playable` interface. Write another class `MusicPlayer` that `implements Playable`. The class `MusicPlayer` represents a music player app to be used in our smartphone. The class `MusicPlayer` has the following private data members: `String artistName, String albumName, String songName, double songDuration` and `int volumeLevel`. The class `MusicPlayer` defines all the abstract methods of `Playable` interface.

Write another interface `Trackable` with the following abstract method: `getCoordinates()`.

Write a class `GPSApp` that `implements Trackable`. The class `GPSApp` represents a location tracker application to be used in our smartphone. The class `GPSApp` has the following private data members: `float xCoordinate, float yCoordinate, int hour` and `int minutes`. The class `GPSApp` defines the method `getCoordinates()`. Write another class `WeatherApp` that `implements Trackable`. The class `WeatherApp` represents a weather condition application to be used in our smartphone. The class `WeatherApp` has the following private data members: `String city, enum currentForecast` and `enum tomorrowsForecast`. The class `WeatherApp` defines the method `getCoordinates()`.

Write a class `Phone`. The class `Phone` has the following private data members: `String model; int modelYear, MusicPlayer player, Radio radio, GPSApp gps` and `WeatherApp weather`.

2. Write a class `SmartphoneApp` that contains the following `public static` methods: `void displayMenu()`, and `void main(String[] args)`. `displayMenu` method will display a sample menu to the user to use the applications in the phone.

In `main`, at first, construct a new `Phone` instance and set the corresponding class members accordingly. You can also create many test instances from `Phone` class by specifying different smartphones. After this part, display the menu options to the user. According to the selection of the user, simulate the necessary actions for particular applications. For example, if the user selects option 1 to use weather application, the program must display a sample weather condition to the user. You can simulate other apps similarly.

Write the program in such a manner that the program reads an integer value from the user among the option numbers in menu. It is expected that the user would enter a number; but what if the user enters a different character or a string? Discuss.