Class: CMSC203 CRN 31414

 Program: Assignment # 2

Instructor: Ahmed Tarek

 Summary of Description: this program is designed to display patient information and procedures based on the arguments passed on to the constructor.

 Due Date: 02/13/2024

 Integrity Pledge: I pledge that I have completed the programming assignment independently.

 I have not copied the code from a student or any source.

Student’s Name: Kebron Ghile

Pseudocode

Start program

Crate Patient class :

* Crate non-static fields(variables) for all elements
* Create a constructor that takes no argument
* Create a constructor that initialize the fields of first, middle and last name using “this” instance in the public constructor.
* Create a constructor that accepts first last and middle name as a parameter and initialize the fields of first, middle and last name using “this” instance in the public constructor.
* Create a constructor that accepts all attributes as an argument and initialize all fields using “this” instance in the public constructor.
* Create a public accessor for all attributes that returns the value of that specific attribute
* Crate a public mutator for all attributes that accepts an argument if we want to change the value of the attribute and doesn’t return a value.
* Crate BuildFullName, a string method that returns full name( all name attributes added with space between)
* Create BuildAddress ,a s string method that returns full address( all address attributes added into a single string)
* Crate buildEmergencyContact , a method that accepts emergency contact attributes and returns a concatenation of both parameters with the phone number formatted using subString.
* Create a toString method that returns fullName, Address ,and EmergencyContactInfo as a single string.

Crate Procedure class:

* Crate non-static fields(variables) for all elements
* Create a constructor that takes no argument
* Create a parameterized constructor that takes procedure name and date as arguments and initialize the fields procedureName and ProcedureDate using “this” instance in the public constructor.
* Create a parameterized constructor that takes all procedure attributes and initialize the particular fields using “this” instance in the public constructor.
* Crate an accessor for every attribute and return their attribute value.
* Crate a mutator for every attribute and change the values using argument if needed to be changed.
* Crate a toString method that returns all procedure attributes formatted in a single String with indentation.

Crate public PatientDriverApp class

* Crate a public method displayPatient that accepts an object argument
  + Display by calling toString method using the class object.
* Crate a public method displayProcedure that accepts an object argument
  + Display by calling toString method using the class object.

Crate main class

* Crate an object for the Patient class and call the constructor with all attributes
* Crate an object for the Procedure class and call the constructor with all attributes
  + Call displayPatient method.
  + Call displayProcedure method as many times as the procedures conducted.
  + For every procedure get the price of procedure by using the object and the accessor then convert the value to integer
  + Crate a total price variable add all the costs of procedures
  + Format and display total
  + Display programer name
  + Display mc number
  + display due date
  + end program