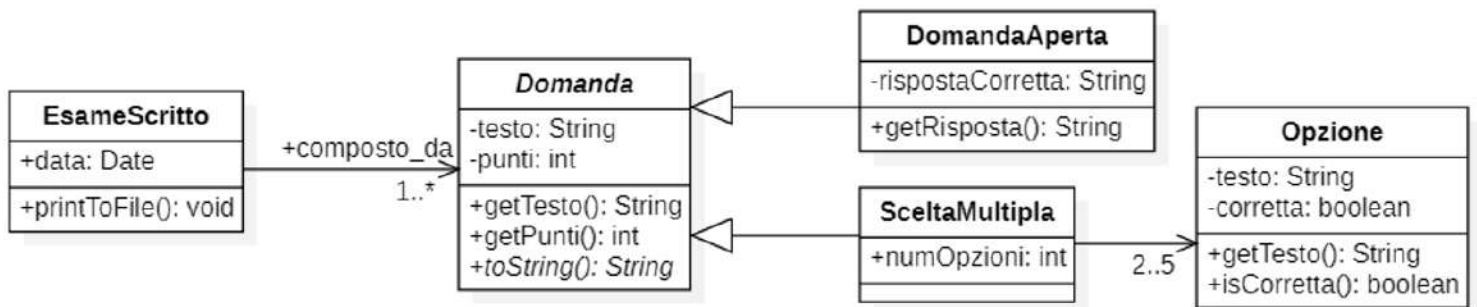


Si scriva tutto il codice Java che è possibile desumere dal seguente class diagram.



Si vuole realizzare un sistema per la prenotazione di prestazioni sanitarie. In particolare, un paziente potrà prenotare una visita specificando il tipo di trattamento, il giorno e l'orario desiderato. Ogni prestazione sanitaria è caratterizzata da un nome, una lista di slot temporali disponibili (su base settimanale) ed una lista di prenotazioni. Il sistema dovrà consentire la prenotazione della prestazione e, nel caso lo slot selezionato non fosse disponibile, dovrà offrire la possibilità di selezionare uno slot disponibile dalla lista degli slot temporali disponibili.

Definire un Class Diagram per modellare la funzionalità di prenotazione di una prestazione, inteso come modello di dominio.

Si vuole realizzare un sistema per la prenotazione di prestazioni sanitarie. In particolare, un paziente potrà prenotare una visita specificando il tipo di trattamento, il giorno e l'orario desiderato. Ogni prestazione sanitaria è caratterizzata da un nome, una lista di slot temporali disponibili (su base settimanale) ed una lista di prenotazioni. Il sistema dovrà consentire la prenotazione della prestazione e, nel caso lo slot selezionato non fosse disponibile, dovrà offrire la possibilità di selezionare uno slot disponibile dalla lista degli slot temporali disponibili.

In relazione all'esercizio precedente, fornire i Mock-up e un Sequence Diagram di analisi della prenotazione di una prestazione sanitaria.

```

Class A {
    ArrayList<Persona> p;
    public void foo() {
        try{
            System.out.println("Length:");
            System.out.println(p.length());
            System.out.println("Done");
        } catch(ArrayIndexOutOfBoundsException e){
            System.out.println("ArrayIndexOutOfBoundsException");
        } catch(NullPointerException e) {
            System.out.println("NullPointerException");
        }
        System.out.println("Exit");
    }
}

```

Si consideri il frammento di codice Java sopra riportato:

- La classe A compila correttamente? (si assuma che tutti gli `import` necessari siano stati correttamente specificati)
- Se la risposta è sì, scrivere l'output di un'invocazione del metodo `foo()`. Se il risultato è il lancio di un'eccezione, specificare di quale eccezione si tratta.
- Se la risposta è no, dire qual è il problema e come si può modificare il codice per farlo compilare.

```

class Vehicle {
    void makeSound(){};
}

class Car extends Vehicle {
    void makeSound(){System.out.println("Vroom");}
}

class Train extends Vehicle {
    void makeSound(){System.out.println("Choo Choo");}
}

```

```

class SomeClass {

    public void foo() {
        Train t = new Train();
        Vehicle v1 = t;
        Vehicle v2 = new Car();

        t.makeSound();
        v1.makeSound();
        v2.makeSound();
    }
}

```

Si consideri il codice Java sopra riportato.

- Il codice compila correttamente? (si assuma che gli import necessari siano definiti) In caso negativo, come si potrebbe modificare il codice affinché la compilazione abbia successo?
- Qual è (considerando anche le eventuali modifiche apportate al punto precedente) l'output su stdout di un'invocazione del metodo `foo()` in `SomeClass`?
- È presente un esempio di polimorfismo? Motivare la risposta. In caso affermativo, indicare in quale/quali parti di codice è presente. In caso negativo, indicare (se possibile) come modificare il codice per introdurne un esempio.