

# Programma per la gestione di una struttura sanitaria.

---

## R1: Pazienti

Il programma funziona attraverso la classe principale **StrutturaSanitaria**. Il programma permette di inserire medici e pazienti all'interno del sistema della sanità. I pazienti sono caratterizzati da nome, cognome e codice fiscale.

Le persone possono essere registrate come pazienti presso il sistema tramite il metodo **aggiungiPersona()** che riceve come parametri nome, cognome e codice fiscale.

Per recuperare le informazioni sui pazienti inseriti si può utilizzare il metodo **getPersona()** che riceve come parametro il codice fiscale e restituisce un oggetto che implementa l'interfaccia **Persona**.

Se non esiste nessuna persona con il codice fiscale specificato, viene generata un'eccezione di **ErrPersonaInesistente**.

Il nome ed il cognome della persona possono essere letti tramite i metodi **getNome()** e **getCognome()** dell'interfaccia **Persona**.

Esempio di persone:

Nome	Cognome	Codice Fiscale
Giovanni	Rossi	RSSGNN33B30F316I
Giuseppe	Verdi	VRDGPP76F09B666I

## R2: Medici

I medici hanno le stesse caratteristiche dei pazienti ed inoltre sono caratterizzati da una matricola. E' possibile registrare un medico presso il sistema tramite il metodo **aggiungiMedico()** che riceve come parametri matricola, nome, cognome e codiceFiscale.

Per recuperare le informazioni sui medici si può utilizzare il metodo **getMedico()** che riceve come parametro la matricola e restituisce un oggetto che implementa l'interfaccia **Medico**. Se non c'è un medico con la matricola specificata presente nel sistema genera un'eccezione di **ErrMedicoInesistente**.

E' possibile conoscere la matricola di un medico tramite il metodo **getMatricola()** dell'interfaccia **Medico**.

## R3: Assegnazione Medici

E' possibile assegnare un medico ad un paziente tramite il metodo **assegnaMedico()** della classe **StrutturaSanitaria** che riceve come parametri la matricola del medico ed il codice fiscale del paziente. Tale metodo genera le opportune eccezioni (**ErrPersonaInesistente**, **ErrMedicoInesistente**) se la matricola o il CF non corrispondono a medici/pazienti inseriti nel sistema.

Deve essere possibile sapere, data una persona, qual'è il medico assegnato tramite il metodo

**getMedico()** dell'interfaccia Persona.

N.B. I medici possono essere a loro volta pazienti di un altro medico.

## R4: Elenco Pazienti

Dato un medico, deve essere possibile ottenere la lista dei pazienti tramite il metodo **getPazienti()** dell'interfaccia Medico.

La lista dei pazienti di un medico deve essere in ordine alfabetico crescente in base a cognome e nome.

## R5: Caricamento da file

Tramite il metodo **caricaDati()**, che riceve come parametro il path di un file, deve essere possibile leggere da file i pazienti ed i medici.

Il file è organizzato a righe, ogni riga può contenere un medico o una persona.

Le righe che contengono una persona iniziano con la lettera "P" seguita da nome, cognome e codice fiscale. Le righe che contengono un medico iniziano con la lettera "M" seguita da matricola, nome, cognome e codice fiscale. Gli elementi di ciascuna riga sono separati gli uni dagli altri tramite il carattere ";".

In caso di errore dei dati presenti su una linea, il metodo deve essere in grado di ignorarla e passare a quella successiva.

In caso di errori di IO il metodo lancia una `IOException`.

---