**TEST CLASSE VISITE**

|  |  |  |  |
| --- | --- | --- | --- |
| **PRECONDIZIONI** | **DESCRIZIONE** | **AZIONI** | **RISULTATO ATTESO** |
| **Test Costruttori** | | | |
|  | Test del costruttore | Creare una visita v1 con id 1, nomeP Davide, cognomeP Carizzoni, nomeM Bob, CognomeM Marley, data e ora della visita uguali a dataOra. | Id=1  NomeP=Davide  CognomeP=Carizzoni  NomeM= Bob  CognomeM=Marley  Data Visita=dataOra |
|  | Test del costruttore di copia | Avere una tessera v1. Creare una tessera copia di v1 e quindi v2=v1 | v2=v1 |
|  | Test del costruttore vuoto | Creare una tessera senza parametri | Id=0  NomeP=  CognomeP=  NomeM=  CognomeM=  Data Visita= null |
| **Test Getter e Setter** | | | |
|  | Get Id | Creare una visita v1 con Id 1, nomeP Davide, cognomeP Carizzoni, nomeM Bob, cognomeM Marley, data visita uguale a dataOra. Usare getId(). | getId=1 |
|  | Get NomeP | Creare una visita v1 con Id 1, nomeP Davide, cognomeP Carizzoni, nomeM Bob, cognomeM Marley, data visita uguale a dataOra. Usare getNomeP (). | getNomeP=”Davide” |
|  | Get CognomeP | Creare una visita v1 con Id 1, nomeP Davide, cognomeP Carizzoni, nomeM Bob, cognomeM Marley, data visita uguale a dataOra. Usare getCognomeP(). | getCognomeP=”Carizzoni” |
|  | Get NomeM | Creare una visita v1 con Id 1, nomeP Davide, cognomeP Carizzoni, nomeM Bob, cognomeM Marley, data visita uguale a dataOra. Usare getNomeM(). | getNomeM=”Bob” |
|  | Get CognomeM | Creare una visita v1 con Id 1, nomeP Davide, cognomeP Carizzoni, nomeM Bob, cognomeM Marley, data visita uguale a dataOra. Usare getCognomeM(). | getCognomeM=”Marley |
|  | Get DataOra | Creare una visita v1 con Id 1, nomeP Davide, cognomeP Carizzoni, nomeM Bob, cognomeM Marley, data visita uguale a dataOra. Usare getDataOra(). | getDataOra=dataOra |
|  | Get VisitaSvolta | Creare una visita v1 con Id 1, nomeP Davide, cognomeP Carizzoni, nomeM Bob, cognomeM Marley, data visita uguale a dataOra, Usare getVisitaSvolta(). | getVisitaSvolta=false |
|  | Set Id | Creare una visita v1 vuota e usare setId(1). | Matricola=1 |
|  | Set NomeP | Creare una visita v1 vuota e usare setNomeP(“Davide”). | NomeP=”Davide” |
|  | Set CognomeP | Creare una visita v1 vuota e usare setCognomeP(“Carizzoni”). | CognomeP=”Carizzoni” |
|  | Set NomeM | Creare una visita v1 vuota e usare setNomeM(“Bob”). | NomeM =”Bob” |
|  | Set CognomeM | Creare una visita v1 vuota e usare setCognomeM(“Marley”). | CognomeM =”Marley” |
|  | Set DataOra | Creare una visita v1 vuota e usare setDataOra(dataOra). | DataOra =dataOra |
|  | Set VisitaSvolta | Creare una visita v1 vuota e usare setVisitaSvolta(false). | VisitaSvolta =false |

**TEST CLASSE NODO**

|  |  |  |  |
| --- | --- | --- | --- |
| **PRECONDIZIONI** | **DESCRIZIONE** | **AZIONI** | **RISULTATO ATTESO** |
| **Test Costruttori** | | | |
|  | Test del costruttore | Creo una visita vuota. Creo un nodo con dentro la visita. | getInfo=tessera  link=null |
|  | Get Info | Creo due visite vuote v1 e v2, creo un nodo contenente v1. Uso set info sul nodo passando la visita v2 usando getInfo(). | nodo.getInfo=v2  nodo.getLink=null |
|  | Get Link | Creo due visite vuote v1 e v2, creo due nodi: uno contenente v1 e l’altro v2. Uso set link sul nodo 1 e gli assegno il link del nodo 2 usando getLink(). | Nodo1.getInfo=v1  Nodo1.getLink=nodo2 |
|  | Set Info | Creo due visite vuote v1 e v2, creo un nodo contenente v1. Uso set info sul nodo passando la visita v1. | nodo.getInfo=v2  nodo.getLnk=null |
|  | Set Link | Creo due visite vuote v1 e v2, creo due nodi: uno contenente v1 e l’altro v2. Uso set link sul nodo 1 e gli assegno il link del nodo 2. | Nodo1.getInfo=v1  Nodo1.getLnk=nodo2 |

**TEST CLASSE LISTAVISITE**

|  |  |  |  |
| --- | --- | --- | --- |
| **PRECONDIZIONI** | **DESCRIZIONE** | **AZIONI** | **RISULTATO ATTESO** |
| **Test Costruttori** | | | |
|  | Test costruttore | Creo una nuova lista di visite vuota | Elementi=0 |
|  | Test costruttore di copia | Creo una nuova lista di visite v2 uguale a v1 | V2=v1 |
| **Getter e setter** | | | |
|  | Get Elementi | Creo una nuova visita e una lista di visite. Inserisco la visita nella lista. | Elementi=1 |
| **Altri metodi** | | | |
|  | **Get Vi**sita | Costruire un oggetto della classe ListaVisite e uno della classe Visite. Aggiungere la visita alla lista e invocare il getVisita in posizione 1, ossia dove è stata aggiunta la visita. | getVisita=v1 |
|  | Get Tessera | Creo una lista l1 e una tessera t1. Inserisco la tessera nella lista e controllo la matricola della tessera nella lista con la matricola della tessera t1 | L1.getTessera(1).getMatricola=  T1.getMatricola() |
|  | Inserisci Visita | Creo una nuova visita e una lista di visita. Inserisco la visita nella lista. | Elementi=1 |
|  | Elimina in Testa | Creo una lista di visita lv1 e una visita v1. Inserisco v1 in lv1 e uso elimina in testa | Elementi=0 |
|  | Elimina in Coda | Creo una lista di visite v1 e una visita v1. Inserisco v1 in lv1 e uso elimina in coda | Elementi=0 |
|  | Elemina in posizione | Creo una lista di visite lv1 e tre visite v1, v2, v3. Inserisci v1, v2, v3 in lv1. Elimino la visita in posizione 2 e poi di nuovo elimino la visita in posizione 2. | Elementi=1 |
|  | Test salvaLaboratorio e caricaLaboratorio | Costruire una ListaVisite ed eseguire la serializzazione. Invocare il metodo con il nome del file con cui lo si vuole salvare. Sucessivamente creare un altro oggetto di tipo ListaVisite dove caricare gli accessi di quel file binario | Lv1=lv2 |
|  | Test esportaLaboratorioCSV | Costruire una ListaVisite. Invocare il metodo con la data con cui lo si vuole salvare. Creare una stringa(visitaCSV) dove inserire l’accesso in forma CSV e confrontare con la stringa letta dal file di testo creato(StringaLetta) | StringaLetta=visitaCSV |