**PIANO DI TEST UTILIZZANDO IL METODO DEL *CATEGORY-PARTITION TESTING* PER LA FUNZIONALITÀ DI “*visualizzazione della lista degli studenti prenotati*”.**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Tipo ricerca** | **Quantità studenti** | **Quantità appelli** | **Quantità corsi** | **Tipo di studenti in Appello** | **Esito ricerca** |
| * Per appello | * 0 studenti [SINGLE] * 1 studente * >1 studenti [prop. +STUDENTE] | * 0 appelli * 1 studente * >1 studenti [prop. +APPELLI] | * 1 corso * >1 corsi [prop. +CORSI] | * Studenti | * Nessuno studente trovato * 1 studente trovato [prop. TROVATO] * >1 studenti trovati [SINGLE][if +STUDENTE] |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Descrizione** | **Classi di equivalenza coperte** | **Pre-condizioni** | **Input** | **Output Attesi** | **Post-condizioni Attese** | **Output Ottenuti** | **Post-condizioni Ottenute** | **Esito**  **(*FAIL*, *PASS*)** |
| 1 | Un corso con un appello e zero studenti prenotati | TIPO RICERCA: **PER APPELLO**  QUANTITÀ CORSI: **1**  QUANTITÀ APPELLI: **1**  QUANTITÀ STUDENTI: **0**  ESITO RICERCA: **0 risultati** | lista\_corsi = {corso1}  lista\_appelli = {appello1}  lista\_studenti = {} | stampaListaStudenti(  appello1) | 0 studenti | appello1.getStudenti() = {} |  |  |  |
| 2 | Un corso con un appello e uno studente prenotato | TIPO RICERCA: **PER APPELLO**  QUANTITÀ CORSI: **1**  QUANTITÀ APPELLI: **1**  QUANTITÀ STUDENTI: **1**  ESITO RICERCA: **1 risultato** | lista\_corsi = {corso1}  lista\_appelli = {appello1}  lista\_studenti = {studente1} | prenotaAppello(studente1, appello1.getId())  stampaListaStudenti(  appello1) | 1 studente | appello1.getStudenti() = {studente1} |  |  |  |
| 3 | Un corso con un appello e più studenti prenotati | TIPO RICERCA: **PER APPELLO**  QUANTITÀ CORSI: **1**  QUANTITÀ APPELLI: **1**  QUANTITÀ STUDENTI: **> 1**  ESITO RICERCA: **2 risultati** | lista\_corsi = {corso1}  lista\_appelli = {appello1}  lista\_studenti = {studente1,studente2} | prenotaAppello(studente1, appello1.getId())  prenotaAppello(studente2, appello1.getId())  stampaListaStudenti(  appello1) | 2 studenti | appello1.getStudenti() = {studente1, studente2} |  |  |  |
| 4 | Un corso con più appelli e uno studente prenotato per ogni appello | TIPO RICERCA: **PER APPELLO**  QUANTITÀ CORSI = **1**  QUANTITÀ APPELLI. = **> 1**  QUANTITÀ STUDENTI: **> 1**  ESITO RICERCA: **2 risultati** | lista\_corsi = {corso1}  lista\_appelli = {appello1, appello2}  lista\_studenti = {studente1,studente2} | prenotaAppello(studente1, appello1.getId())  prenotaAppello(studente2, appello2.getId())  stampaListaStudenti(  appello1)  stampaListaStudenti(  appello2) | 1 studente per ogni appello | appello1.getStudenti() = {studente1}  appello2.getStudenti() = {studente2} |  |  |  |
| 5 | Un corso con più appelli e più studenti prenotati ad un solo appello | TIPO RICERCA: **PER APPELLO**  QUANTITÀ CORSI = **1**  QUANTITÀ APPELLI. = **> 1**  QUANTITÀ STUDENTI: **> 1**  ESITO RICERCA: **2 risultati** | lista\_corsi = {corso1}  lista\_appelli = {appello1, appello2}  lista\_studenti = {studente1,studente2} | prenotaAppello(studente1, appello1.getId())  prenotaAppello(studente2, appello1.getId())  stampaListaStudenti(  appello1) | 2 studenti per il primo appello | appello1.getStudenti() = {studente1, studente2} |  |  |  |
| 6 | Un corso con più appelli e più studenti prenotati a più appelli | TIPO RICERCA: **PER APPELLO**  QUANTITÀ CORSI = **1**  QUANTITÀ APPELLI. = **> 1**  QUANTITÀ STUDENTI: **> 1**  ESITO RICERCA: **3 risultati** | lista\_corsi = {corso1}  lista\_appelli = {appello1, appello2}  lista\_studenti = {studente1, studente2, stuednte3} | prenotaAppello(studente1, appello1.getId())  prenotaAppello(studente2, appello1.getId())  prenotaAppello(studente3, appello2.getId())  stampaListaStudenti(  appello1)  stampaListaStudenti(  appello2) | 2 studenti per il primo appello  1 studenti per il secondo appello | appello1.getStudenti() = {studente1, studente2}  appello2.getStudenti() = { studente3} |  |  |  |
| 7 | Più corsi, più appelli e più studenti prenotati a più appelli | TIPO RICERCA: **PER APPELLO**  QUANTITÀ CORSI = **>** **1**  QUANTITÀ APPELLI. = **> 1**  QUANTITÀ STUDENTI: **> 1**  ESITO RICERCA: **2 risultati** | lista\_corsi = {corso1,corso2}  lista\_appelli = {appello1,appello2}  lista\_studenti = {studente1,studente2} | prenotaAppello(studente1, appello1.getId())  prenotaAppello(studente2, appello2.getId())  stampaListaStudenti(  appello1)  stampaListaStudenti(  appello2) | 1 studente per il primo appello  1 studente per il secondo appello | appello1.getStudenti() = {studente1 }  appello2.getStudenti() = { studente2} |  |  |  |
| 8 | Più corsi, più appelli e uno studente prenotato a più appelli | TIPO RICERCA: **PER APPELLO**  QUANTITÀ CORSI = **>** **1**  QUANTITÀ APPELLI. = **> 1**  QUANTITÀ STUDENTI: **1**  ESITO RICERCA: **0 risultati** | lista\_corsi = {corso1,corso2}  lista\_appelli = {appello1,appello2}  lista\_studenti = {studente1} | prenotaAppello(studente1, appello1.getId())  prenotaAppello(studente1, appello2.getId())  stampaListaStudenti(  appello1)  stampaListaStudenti(  appello2) | 1 stuente prenotato a più appelli | appello1.getStudenti() = {studente1}  appello2.getStudenti() = { studente1} |  |  |  |
| 9 | Un corso, un appello, uno studente che si prenota due volte allo stesso appello | TIPO RICERCA: **PER APPELLO**  QUANTITÀ CORSI = **1**  QUANTITÀ APPELLI. = **1**  QUANTITÀ STUDENTI: **1**  ESITO RICERCA: **1 risultati** | lista\_corsi = {corso1}  lista\_appelli = {appello1}  lista\_studenti = {studente1} | prenotaAppello(studente1, appello1.getId())  prenotaAppello(studente1, appello1.getId())  stampaListaStudenti(  appello1) | 1 studente prenotato | appello1.getStudenti() = {studente1} |  |  |  |
| 10 | Un corso, zero appelli, uno studente | TIPO RICERCA: **PER APPELLO**  QUANTITÀ CORSI = **1**  QUANTITÀ APPELLI. = **0**  QUANTITÀ STUDENTI: **1**  ESITO RICERCA: **0 risultati** | lista\_corsi = {corso1}  lista\_appelli = {}  lista\_studenti = {studente1} | prenotaAppello(studente1, appello1.getId())  stampaListaStudenti(  appello1) | 0 studenti prenotati | appello1.getStudenti() = {} |  |  |  |
| 11 | Un corso, un appello, più studenti di cui uno si prenota due volte allo stesso appello | TIPO RICERCA: **PER APPELLO**  QUANTITÀ CORSI = **1**  QUANTITÀ APPELLI. = **1**  QUANTITÀ STUDENTI: **> 1**  ESITO RICERCA: **0 risultati** | lista\_corsi = {corso1}  lista\_appelli = {appello1}  lista\_studenti = {studente1,studente2} | prenotaAppello(studente1, appello1.getId())  prenotaAppello(studente1, appello1.getId())  prenotaAppello(studente2, appello1.getId())  stampaListaStudenti(  appello1) | 2 studenti prenotati | appello1.getStudenti() = {studente1, studente2} |  |  |  |
| 12 | Stampa lista studenti prenotati senza specificare nessun appello | TIPO RICERCA: **PER APPELLO**  QUANTITÀ CORSI = **> 1**  QUANTITÀ APPELLI. = **> 1**  QUANTITÀ STUDENTI: **> 1**  ESITO RICERCA: **0 risultati** | lista\_corsi = {corso1, corso2}  lista\_appelli = {appello1, appello2}  lista\_studenti = {studente1,studente2, studente 3} | prenotaAppello(studente1, appello1.getId())  prenotaAppello(studente2, appello1.getId())  prenotaAppello(studente3, appello2.getId())  stampaListaStudenti(null) | Stampa la lista degli studenti prenotati di tutti gli appelli | appello1.getStudenti() = {studente1, studente2}  appello2.getStudenti()  = {studente3} |  |  |  |