# Interfacce della classe

3.Pilota

|  |  |
| --- | --- |
| Nome Classe | Pilota |
| Descrizione | La classe fornisce tutte le informazioni necessarie per la gestione del pilota registrato al sistema. |
| Metodi | + getPunteggio():int  + setPunteggio(int punteggio):void  +getNumeroVittorie():int  +setNumeroVittorie(int numeroVittorie) : void  +getNumeroPole() :int  +setNumeroPole(int numeroPole):void  +getNumeroPiazzamenti() : int  +setNumeroPiazzamenti(int numeroPiazzamenti) : void  +getNumeroRitiri(): int  +setNumeroRitiri(int numeroRitiri):void |

|  |  |
| --- | --- |
| Nome Classe | Pilota |
| Nome Metodo | + setPunteggio(int punteggio):void |
| Descrizione | Il metodo setta il punteggio del pilota |
| Pre-Condizioni | **context**: Pilota :: setPunteggio(int punteggio):void  **pre:** (punteggio>=0) |
| Post-Condizioni | - |
| Invarianti | **context** Pilota **inv**:0<=self.punteggio <=575 |

|  |  |
| --- | --- |
| Nome Classe | Pilota |
| Nome Metodo | + setNumeroVittorie(int numeroVittorie):void |
| Descrizione | Il metodo setta il numero di vittore del pilota |
| Pre-Condizioni | **context**: Pilota :: setNumeroVittorie(int numeroVittorie):void  **pre:** (numeroVittorie >=0) |
| Post-Condizioni | - |
| Invarianti | **context** Pilota **inv**:0<=self.numeroVittorie <=23 |

|  |  |
| --- | --- |
| Nome Classe | Pilota |
| Nome Metodo | + setNumeroPole(int numeroPole):void |
| Descrizione | Il metodo setta il numero di pole effettuate dal pilota |
| Pre-Condizioni | **context**: Pilota :: setNumeroPole(int numeroPole):void  **pre:** (numeroPole >= 0) |
| Post-Condizioni | - |
| Invarianti | **-** |

|  |  |
| --- | --- |
| Nome Classe | Pilota |
| Nome Metodo | +setNumeroPiazzamenti(int numeroPiazzamenti) : void |
| Descrizione | Il metodo setta il numero di piazzamenti effettuati dal pilota |
| Pre-Condizioni | **context**: Pilota :: setNumeroPiazzamenti(int numeroPiazzamenti) : void  **pre:** (numeroPiazzamenti >=0) |
| Post-Condizioni | - |
| Invarianti | **-** |

|  |  |
| --- | --- |
| Nome Classe | Pilota |
| Nome Metodo | + setNumeroRitiri(int numeroRitiri):void |
| Descrizione | Il metodo setta il numero di ritiri effettuati dal pilota |
| Pre-Condizioni | **context**: Pilota :: setNumeroRitiri(int numeroRitiri):void  **pre:** (numeroRitiri >=0) |
| Post-Condizioni | - |
| Invarianti | **context** Pilota **inv**:0<=self.numeroRitiri <=23 |

4.PilotaDAO

|  |  |
| --- | --- |
| Nome Classe | PilotaDAO |
| Descrizione | La classe gestisce l’interazione tra la classe pilota e il database. |
| Metodi | + doSave( Pilota pilota): void  + doRetrieveByPilota(Pilota pilota):Pilota |

5.Tecnico

|  |  |
| --- | --- |
| Nome Classe | Tecnico |
| Descrizione | La classe fornisce tutte le informazioni necessarie per la gestione del tecnico registrato al sistema. |
| Metodi | +getSetupList(): Collection<Setup>  + setSetupList(Collection<Setup> setupList):void  + addSetup(Setup s):void  + removeSetup(Setup s):void |

|  |  |
| --- | --- |
| Nome Metodo | + setSetupList(Collection<Setup> setupList):void |
| Descrizione | Il metodo setta il numero totale di setup per il tecnico. |
| Pre-Condizioni | **context**: Pilota ::setSetupList(Collection<Setup> setupList):void  **pre:** (setupList != null) |
| Post-Condizioni | - |
| Invarianti | **-** |

|  |  |
| --- | --- |
| Nome Classe | Tecnico |
| Nome Metodo | + addSetup(Setup s):void |
| Descrizione | Il metodo aggiunge un nuovo setup per il tecnico |
| Pre-Condizioni | **context**: Tecnico :: addSetup(Setup s):void  **pre:** (s != null) |
| Post-Condizioni | - |
| Invarianti | **-** |

|  |  |
| --- | --- |
| Nome Classe | Tecnico |
| Nome Metodo | + removeSetup(Setup s):void |
| Descrizione | Il metodo rimuove un setup per il tecnico |
| Pre-Condizioni | **context**: Tecnico :: removeSetup(Setup s):void  **pre:** (s != null) |
| Post-Condizioni | - |
| Invarianti | **-** |