it unimal managaly an it.unimol.monopoly Giocatore -nome: String
-id: Int
-denaro: Int
-contrattlPosseduti: List
-numContratti: Int
-isPrigioniero: boolean
+aux Main +menu: MenuPrincipale +main(): void GestoreGiocatori -giocatori: Map<Giocatore> +GestoreGiocatori(): <constructor> +aggiungiGiocatore(int id, Giocatore giocato +toString(): String it.unimol.monopoly.cli Gioco MenuPrincipale +Gloco(GestoreGlocatori glocatori): <constructor>
*setDenaroGlocatori(GestoreGlocatori glocatori): void
*setMmcContrattiliniziali(GestoreGlocatori glocatori): void
*generaOrdine(GestoreGlocatori glocatori): void
*setCslocatoreInizlate(GestoreGlocatori glocatori): void
*setGlocatoreInizlate(GestoreGlocatori glocatori): Glocatore MenuPrincipale
+menuTurno: MenuTurno
+menuPrigione: MenuPrigione
+giocatori: GestoreGiocatori
-contratti: GestoreContratti
+turno: Turno
-gameStarted: boolean
-vuoleUscire: boolean
-scelta: int Contratto -nome: String -id: int +setContrattiPartita(): GestoreContratti
+generaContratti(GestoreContratti contratti): List<Contratto
+compraContrattiIniziali(Giocatore giocatore, List contratti): int -id: int -prezzo: int -affitto: int -isDisponibile: -aux +esegui(): void +mostraMenu(): void +gestisciInput(): boolean +nuovoGiocatore(): void +vediGiocatori(): void *Contratto(String nome, int prezzo, int affitto): <constructor*acquistaContrationiziale(Giocatore giocatore, Contratto contratto): boolean
*acquistaContratind(Giocatore giocatore, Contratto contratto): boolean
*atrostrang(): String
*disString(): String(): String
*disString(): String
*disString(): String
*disString(): String(): +avviaGioco(): void +confermaUscita(): bool pagaAffitto(Giocatore giocatore1, Giocatore giocatore2, Contratto contratto): boolea GestoreContratti Tassa -valore: int +Tassa(int valore): <constructor> +aggiungiContratto(int id, Contratto contratto): void +toString(): String +esegui(Giocatore giocatore, Turno turno, GestoreGiocatori giocatori, GestoreContratti contratti): void +mostraMenu(): void +esetsicilinu(Giocatore giocatore. Turno turno. GestoreGiocatori giocatori. GestoreContratti contratti): boolean -gestiscilinput(Gocastere giocatore, Tumo tumo, GestoreGiocatori giocatore strampaControl(coatore giocatore) costare vi satismpaControl(coatore giocatore). GestoreContratti contratti) void visualatizaContratti (Coatore giocatore). Void vipaquaMitto(Ciocatore giocatore, GestoreGiocatori giocatori): void vipaqiamassi(Ciocatore giocatore, CestoreGiocatori giocatori): void vivaqiamassi(Ciocatore giocatore, Tumo tumo): void vivaqiamassi(Ciocatore giocatore, Ciocatore, Vivaquiamassi(Ciocatore)): void vivaqiamassi(Ciocatore): voi Prigione MenuPrigione -tassaUscita: final int -prigione: Prigione -recScelta -vuoleUscire: boolean +esciGratis(Giocatore giocatore): void +esciPagando(Giocatore giocatore, Turno turno, GestoreGiocatori giocatori): boolean +resta(Giocatore giocatore, Turno turno): void -scelta: int Turno +esegui(Giocatore giocatore, MenuTurno menuTurno, Turno turno, GestoreGiocatori giocatori, GestoreContratti contratti): void +mostraMenu(): void +playerId: int +endTurno(Giocatore giocatore, Turno turno): void passaDalVla(Giocatore giocatore): void vaiInPrigione(Giocatore giocatore): void terminaTurno(Giocatore giocatore): boole