

# Qwirkle

C

Joueur

- pseudo : string
- scoreTot : int
- scoreTour : int
- noTour : int
- roundPlayed : int
- mainTuile : Combinaison
- playing : Combinaison
- nbJoueurs : int
- playerList : List<Joueur>
- currentPlayer : Joueur

- Joueur()
- Joueur(pseudo : string)
- setPseudo(nom : string) : void
- setScoreTot(score : int) : void
- setNoTour(value : int) : void
- setMainTuile(tuiles : Combinaison) : void
- setScoreTour(value : int) : void
- setRoundPlayed(nb : int) : void
- getPlaying() : Combinaison
- getPseudo() : string
- getScoreTot() : int
- getScoreTour() : int
- getMainTuile() : Combinaison
- getNoTour() : int
- getRoundPlayed() : int
- setPlaying(combi : Combinaison) : void
- giveUp(pioche : Pioche) : void
- play() : void
- countGamePoints(pioche : Pioche) : void
- setNbJoueurs(number : int) : void
- setPlayerList(list : List<Joueur>) : void
- setCurrentPlayer(player : Joueur) : void
- getNbJoueurs() : int
- getPlayerList() : List<Joueur>
- getCurrentPlayer() : Joueur
- chooseOrder(tab\_joueur : List<Joueur>) : List<Joueur>
- chooseRanking(tab\_joueurs : List<Joueur>) : List<Joueur>

C

Tuile

- couleur : string
- forme : string
- placable : bool
- detenteur : Joueur
- line : int
- column : int

- setPlacable(value : bool) : void
- setDetenteur(joueur : Joueur) : void
- setLine(id\_line : int) : void
- setColumn(id\_column : int) : void
- getCouleur() : string
- getForme() : string
- getPlacable() : bool
- getDetenteur() : Joueur

C

Plateau

- nbLines : int
- nbColumns : int
- gridPlat : List<List<Tuile>>
- nbTuiles : int

- setNbLines(number : int) : void
- setNbColumns(number : int) : void
- initGrid() : List<List<Tuile>>
- setGrid(grid : List<List<Tuile>>) : void
- setNbTuiles(number : int) : void
- getNbLines() : int
- getNbColumns() : int
- getGrid() : List<List<Tuile>>
- getNbTuiles() : int
- addLine(index : int) : void
- removeLine(index : int) : void
- addColumn(index : int) : void
- removeColumn(index : int) : void
- addTuile(tuile : Tuile, line : int, column : int) : void
- removeTuile(tuile : Tuile, line : int, column : int) : void
- reset() : void
- checkAlignment(id\_line : int, id\_column : int, tuile\_to\_add : Tuile) : bool

C

Pioche

- Pioche()
- Pioche(tuiles : List<Tuile>)
- giveTuilesToPlayer(player : Joueur, amount : int) : Combinaison
- echangeTuile( echange : Combinaison ) : Combinaison

C

Combinaison

△ tuiles : List<Tuile>

- Combinaison()
- Combinaison(tuiles : List<Tuile>)
- setTuiles(tuiles : List<Tuile>) : void
- getTuiles() : List<Tuile>
- addTuiles(tuile : Tuile) : void
- removeTuiles(tuile : Tuile) : void

