+ Game( startingLevel : int )

+ runLevel( numLevel : int ) : void

+ getLevel() : int

+ setLevel( newLevel : int ) : void

+ getScore() : int

+ setScore( newScore : int ) : void

+ getLives() : int

+ setLive( newLives : int ) : void

+ getColor() : Color

+ restartLevel( b : Board ) : void

+ levelScore( timeLeft : int ) : int

-score : int

-frogDirection : frogDirection

-colors : HashMap

-level : int

-lives : int

Class Board

-rows : ArrayList<Row>

-numRows : int

-numFrames : int

-rowSize : int

-frogCurrentRow : int

-frogXIndex : int

-boardScore : int

Class GameLoop

Class Row

+ Board( emptyRowCoef : double, rowSpeedCoef : int, numEnemiesCoef : int )

+ buildBoard( emptyRowCoef : double, rowSpeedCoef : int, numEnemiesCoef : int ) : void

+ boardShift( frogShift : int ) : void

+ drawBoardString( level : int, score : int, lives : int ) : String

+ drawBoard() : void

+ completeBoard() : void

+ moveFrog( moveCode : int ) : void

+ setFrogRow( row : int ) : void

+ setFrogXIndex( x : int ) : void

+ setFrog() : void

+ removeFrog() : void

+ getBoardScore() : int

+ setBoardScore( newScore : int ) : void

-units : ArrayList<Unit>

-rowSpeed : int

-numEnemies : int

-rowSize : int

+ Row()

+ shouldShift( numFrames : int) : boolean

+ rowShift( numFrames : int, frogX : int) : void

+ toString() : String

+ setEnemy( x : int ) : void

+ moveFrogLeftRight( lr : int, frogIndex : int) : void

+ hasEnemy( x : int ) : Boolean

+ frogLeaves( x : int ) : void

+ frogAppears( x : int ) : void

Class Player extends Unit

-sprite : String

+ getSprite() : String

Abstract Class Unit

+ toString() : String

+ abstract getSprite() : String

-sprite : String

+ getSprite() : String

Class Enemy extends Unit

-sprite : String

+ getSprite() : String

Class Ground extends Unit