Color Switch: Class UML Diagram

Naval Kumar Shukla (2019065) || Abhishek Chaturvedi (2019401)

Java.Lang.Exception

PlayerNotFoundException
- + PlayerNotFoundEception()

HomeScreen
- currentPlayer: Player
- playerList: HashMap<String,Player>

- + newGame(): void
- + resumeGame(): void
- + exitGame(): void
- + serialise(): void
- + deserialise(): void
- + main(args: String[]): void

ClassNotFoundException

IOException

<<interface>> Serializable

GameElements
- # position: float[]
- # dimensions: float[]

- + getPosition(): float[]
- + getDimension(): float[]
- + setPostion(pos: float[])
- + setDimension(dim: float[])

Player
- NAME: String
- gameScreen: GameScreen
- score: long

- + Player(String name)
- + getName(): String
- + getScreen(): GameScreen
- + getScore(): long
- + newGame(): void
- + pauseGame(): void

InsufficientScoreException
- + InsufficientScoreException()

Ball
- JUMP_VALUE: double
- fall: boolean
- fallSpeed: double
- jumpSpeed: double

- + Ball(pos: float[], dim: float[], fSpeed: double, jSpeed: double)
- + getJumpValue(): double
- + getFallSpeed(): double
- + getJumpSpeed(): double
- + isFalling(): boolean
- + jump(): void
- + setFallSpeed(xSpeed: double)
- + setJumpSpeed(xSpeed: double)
- + disappear(): void

ColorSwitcher
- + ColorSwitcher(pos: float[], dim: float[])
- + pickColor(): int[]
- + disappear(): void

Star
- point: int
- + Star(pos: float[], dim: float[], p: int)
- + getPoint(): int
- + setPoint(p: int)
- + disappear(): void

Obstacle
- duration: double
- angle: double

- + getDuration(): double
- + getAngle()
- + setDuration(d: double)
- + move(): void

AbstractLineObstacle
- + move(): void

OneDLineObstacle
- + OneDLineObstacle(pos: float[], dim: float[], dur: double, xAngle: double)
- + move(): void

TwoDLineObstacle
- + TwoDLineObstacle(pos: float[], dim: float[], dur: double, xAngle: double)
- + move(): void

CircleObstacle
- + CircleObstacle(pos: float[], dim: float[], dur: double, xAngle: double)
- + move(): void

EllipseObstacle
- + EllipseObstacle(pos: float[], dim: float[], dur: double, xAngle: double)
- + move(): void

PolygonObstacle
- + move(): void

RectangleObstacle
- + RectangleObstacle(pos: float[], dim: float[], dur: double, xAngle: double)
- + move(): void

OctagonObstacle
- + OctagonObstacle((pos: float[], dim: float[], dur: double, xAngle: double)
- + move(): void

HexagonObstacle
- + HexagonObstacle((pos: float[], dim: float[], dur: double, xAngle: double)
- + move(): void

<<interface>> Animation
- + disappear(): void

GameScreen
- starScore: int
- currentlevel: Level
- SCREEN_SIZE: int[]
- ball: Ball
- obstacleList: ArrayList<Obstacle>
- colorSwitchers: ArrayList<ColorSwitcher>
- stars: ArrayList<Star>

- + GameScreen()
- + getScore(): int
- + getLevel(): Level
- + getScreenSize(): int[]
- + getBall(): Ball
- + getObstaclesList(): ArrayList<Obstacle>
- + getColorSwitchers(): ArrayList<ColorSwitcher>
- + getStars(): ArrayList<Star>
- + updateLevel(): void
- + nextObstacle(): void
- + loseGame(): void

Level
- levelNumber: int
- revivalScore: int
- ballFallSpeed: double
- ballJumpSpeed: double
- obstacleDuration: double
- availableObstacles: ArrayList<String>

- + Level(int levelNumber)
- + getLevelNumber(): int
- + getRevivalScore(): int
- + getFallSpeed(): double
- + getJumpSpeed(): double
- + getObstacleDuration(): double
- + getAvailableObstacles(): ArrayList<String>

Relationship labels: <<extends>>, <<throws>>, <<implements>>, <<extend>>