

<i>AbstractCharacter</i>
- SIZE: int - x: int - y: int - pixelsPerStep: int
+ AbstractCharacter(x: int, y: int, pixelsPerStep: int) + move(move: Move): void + moveBack(currentDirection: Move): void + getSize(): int + getX(): int + getY(): int + getCollIndex(): int + getRowIndex(): int

Inherits

<i>Enemy</i>
- currentDirection: Move
+ Enemy(x: int, y: int, vertical: boolean) + changeDirection() + getCurrentDirection(): Move - randomDirection(vertical: boolean): Move

<i>Player</i>
- PLAYER_START_X: int - PLAYER_START_Y: int - PLAYER_PIXELS_BY_STEP: int - explosionRadius: int - bombCount: int - floor: Floor - up: Action - right: Action - down: Action - left: Action - dropBomb: Action
+Player(bombermanComponent: BombermanComponent, floor: Floor) +setPlayerButtons(bombermanComponent: BombermanComponent) +getBombCount(): int +setBombCount(bombCount: int): void +getExplosionRadius(): int +setExplosionRadius(explosionRadius: int): void -movePlayer(move: Move): void

<i>Engine</i>
- TIME_STEP: int - width: int - height: int - nrOfEnemies: int - clockTimer: Timer
+ main(args: String[]): void + startGame(): void + gameOver(frame: BombermanFrame, floor: Floor): void + tick(frame: BombermanFrame, floor: Floor): void

<i>Bomb Radius PU</i>
- rowIndex: int - collIndex: int
+ BombRadiusPU() + addToPlayer(Player player) + getName(): String

<i>Explosion</i>
- rowIndex: int - collIndex: int - duration: int
+ Explosion(rowIndex: int, collIndex: int) + getRowIndex(): int + getCollIndex(): int + getDuration(): int + setDuration(duration: int): void

<i>AbstractPowerup</i>
- POWERUP_SIZE: int - x: int - y: int - name: String
+ AbstractPowerup(x: int, y: int) + addToPlayer(player: Player): void + getPowerupSize(): int + getX(): int + getY(): int + getName(): String

Inherits

<i>Bomb CounterPU</i>
+ BombCounterPU(rowIndex: int, collIndex: int) + addToPlayer(player: Player): void + getName(): String

<i>Bomberman Component</i>
- SQUARE_SIZE: int - CHARACTER_ADJUSTMENT_FOR_PAINT: int - SQUARE_MIDDLE: int - BOMB_ADJUSTMENT_1: int - BOMB_ADJUSTMENT_2: int - PAINT_PARAMETER_13: int - PAINT_PARAMETER_15: int - PAINT_PARAMETER_17: int - PAINT_PARAMETER_18: int - PAINT_PARAMETER_19: int - PAINT_PARAMETER_20: int - PAINT_PARAMETER_24: int - floor: Floor - colorMap: AbstractMap<FloorTile, Color>
+ BombermanComponent(floor: Floor) + getSquareSize(): int + getSquareMiddle(): int + getPreferredSize(): Dimension + floorChanged(): void + paintComponent(g: Graphics): void - paintBreakableBlock(rowIndex: int, collIndex: int, g2d: Graphics): void + paintPlayer(player: Player, g2d: Graphics2D): void + paintEnemy(e: Enemy, g2d: Graphics2D): void + paintPowerups(p: AbstractPowerup, g2d: Graphics2D): void + paintBombs(b: Bomb, g2d: Graphics2D): void + paintExplosions(tup: Explosion, g2d: Graphics2D): void

Inherits

Floor Listener
-floorChanged()

<i>Bomb</i>
- BOMBSIZE: int - STARTCOUNTDOWN: int - timeToExplosion: int - rowIndex: int - collIndex: int - explosionRadius: int - playerLeft: boolean
+ Bomb(rowIndex: int, collIndex: int, explosionRadius: int) + getRowIndex(): int + getCollIndex(): int + getBOMBSIZE(): int + getTimeToExplosion(): int + setTimeToExplosion(timeToExplosion: int): void + getExplosionRadius(): int + isPlayerLeft(): boolean + setPlayerLeft(playerLeft: boolean): void

<i>Floor</i>
- CHANCE_FOR_BREAKABLE_BLOCK: double - CHANCE_FOR_RADIUS_POWERUP: double - CHANCE_FOR_COUNTER_POWERUP: double - tiles: FloorTile[][] - width: int - height: int - floorListeners: Collection<FloorListener> - player: Player - enemyList: Collection<Enemy> - bombList: List<Bomb> - powerupList: Collection<AbstractPowerup> - explosionList: Collection<Bomb> - explosionCoords: Collection<Explosion> - isGameOver: boolean
+Floor(width: int, height: int, nrOfEnemies: int) +getFloorTile(rowIndex: int, collIndex: int): FloorTile +getWidth(): int +getHeight(): int +getPlayer(): Player +getEnemyList(): Collection<Enemy> +getBombList(): Iterable<Bomb> +getBombListSize(): int +getPowerupList(): Iterable<AbstractPowerup> +getExplosionCoords(): Iterable<Explosion> +getIsGameOver(): boolean +setIsGameOver(value: boolean): void +addToBombList(bomb: Bomb): void +createPlayer(bombermanComponent: BombermanComponent, floor: Floor): void +squareToPixel(squareCoord: int): int +moveEnemies(): void +addFloorListener(bl: FloorListener): void +notifyListeners(): void +bombCountdown(): void +explosionHandler(): void +playerInExplosion(): void +enemyInExplosion(): void +characterInExplosion(): void +collisionWithEnemies(): boolean +collisionWithBombs(abstractCharacter: AbstractCharacter): boolean +collisionWithBlock(abstractCharacter: AbstractCharacter): boolean +collisionWithPowerup(): void +squareHasBomb(rowIndex: int, collIndex: int): boolean +checkIfPlayerLeftBomb(): void +placeBreakable(): void +clearSpawn(): void +spawnPowerup(rowIndex: int, collIndex: int): void +placeUnbreakableAndGrass(): void +spawnEnemies(nrOfEnemies: int): void +collidingCircles(abstractCharacter: AbstractCharacter, x: int, y: int): boolean +squareCircleInstersect(row: int, col: int, abstractCharacter: AbstractCharacter): boolean