

Stock

PicturesStock
<ul style="list-style-type: none">- pictures: Vector<Imagelcon>- point2: Point- cat_5x5: Imagelcon []- cat_4x4: Imagelcon []- cat_3x3: Imagelcon []- cyber_5x5: Imagelcon []- cyber_4x4: Imagelcon []- cyber_3x3: Imagelcon []- sushi_5x5: Imagelcon []- sushi_4x4: Imagelcon []- sushi_3x3: Imagelcon []
<ul style="list-style-type: none">+ PicturesStock(): void+ initPictures(): void+ getPics(): Vector<Imagelcon>+ getGamePictures(int size, String pictureName): Imagelcon []+ getPics3(String pictureName): Imagelcon []+ getPics4(String pictureName): Imagelcon []+ getPics5(String pictureName): Imagelcon []

Boards
<ul style="list-style-type: none">- boards3x3: Vector<Integer>- boards4x4: Vector<Integer>- boards5x5: Vector<Integer>
<ul style="list-style-type: none">+ Boards (): void- initCSVFile(): void+ getBoard(int size): int []+ getBoards3x3Size(): int+ getBoards4x4Size(): int+ getBoards5x5Size(): int

SplitImage

Game Tools

Board
<ul style="list-style-type: none">- N: final int- M: final int- imageIndex: int- StringImage: String- stack: MovesStack- array: Point [][]
<ul style="list-style-type: none">+ Board (int sizeM, int sizeN, PicturesStock pics ,int imageIndex, Boards boards): void- initRandomArrayIcons (): void- initBoardMatrix(Boards boards): void- initArrayIcons(PicturesStock pics): void+ searchPeace (int way): boolean- checkNeighbors (int x, int y, int way): Point+ makeStep(int x, int y): boolean- switchPeaces (Point one , Point two): void+ undo (): void+ isGameOver(): boolean+ getPoint(int i, int j): Point+ getColLength (): int+ getRowLength (): int

GameOver extends JFrame implements ActionListener
<ul style="list-style-type: none">- lblWinner: JLabel- lblSteps: JLabel- lblTime: JLabel- lblFireworks: JLabel- btnExit: JButton
<ul style="list-style-type: none">+ GameOver(int steps, String time, ImageIcon im): void

Game extends JFrame implements ActionListener
<ul style="list-style-type: none">+ ONE_SECOND: final inttimer: Timer- Pics: PicturesStock- BoardStock: Boards- imageIndex: int- isGameOver: boolean- counter: int- Steps: int- steps: JLabel- btnUndo: JButton- groupLayout: GroupLayout- board: Board- btnBoard: JButton [][]- ROWS: final int- COLS: final int
<ul style="list-style-type: none">+ Game(int ROWS,int COLS, int imageIndex, PicturesStock pics, Boards boardStock): void+ actionPerformed(ActionEvent arg0): void- paintBoard(): void- paintGameOver (): void- InitializeBoard(): void+ getTime(int numSec): String

Menus

PictureAndSize extends JFrame implements ActionListener
<ul style="list-style-type: none">- pics: PicturesStock- boardStack: Boards- lblGamesSize: JLabel- b3: JRadioButton- b4: JRadioButton- b5: JRadioButton- buttonGroup: final ButtonGroup- lblGamesPicture: JLabel- scrollPane: JScrollPane- chosenImage: JButton- list: JList- btnReturn: JButton- btnStartGame: JButton
<ul style="list-style-type: none">+ PictureAndSize(PicturesStock pics, Boards boardStack): void

GameMenu extends JFrame implements ActionListener
<ul style="list-style-type: none">+ GameMenu (PicturesStock pics, Boards boards): void+ main(String[] args): void

Point

Move
- point1: Point - point2: Point - next: Move - prev: Move
+ Move(Point p1, Point p2): void + setNext (Move m): void + setPrev (Move m): void + setPoint1 (Point p): void + setPoint2 (Point p): void + getNext (): Move + getPrev (): Move + getPoint1(): Point + getPoint2(): Point

MovesStack
- head: Move - tail: Move
+ MovesStack (): void + MovesStack (Move m): void + push(Move m): void + pop(): Move

Point
- imageIndex: int - icon: ImageIcon - M: int - N: int - x: int - y: int
+ Point(int M, int N, int index): void + pointChanged (int imageIndex, ImageIcon icon): void + isFinalPosition (int i, int j): boolean + getIndex(): int + setIndex(int i): void + getIcon(): ImageIcon + setIcon(ImageIcon icon): void + setX(int x): void + getX(): int + setY(int y): void + getY(): int