

### CtrlDomain **Domain Controllers** - data: CtrlData currentKakuro: Kakuro + CtrlDomain() CtrlGenerate + startNewGame(int difficulty, int kakuroSizeRow, int kakuroSizeColumn) - currentKakuro: Kakuro + setCorrectValues() + isFinished() + CtrlGenerate() + checkValidity(int x, int y, int value) + setKakuro(Kakuro kakuro) + helpMyValue(int x, int y) + countWhiteCellsV(Cell[][] board, int i, int j) + helpCorrectNumber(int x, int y) + countWhiteCellsH(Cell[][] board, int i, int j) + validate() + ineCellsRow(Cell[][] board, int[][][] tempBoard) + resolve() + nineCellsCol(Cell[][] board, int[][][] tempBoard) + generate(int size, int diff) + computePosSumsRec(int arr[], int index, int num, int reduced + kakuroSetValue(int x, int y, int value) + computePosSums(int x, int n, int no) + getRowSize() + allZero(int[][][] tempBoard, int i, int j) + getColumnSize() + intersection3(int a[], int b[]) + getKakuroToString() + intersection2(int a[], int b[]) + getCorrectKakuroToString() + intersection(int[] valuesH, int[] valuesV) + searchKakuro(int difficulty, int kakuroSizeRow, int kakuroSizeColumn) + isUnique(int[] a) + getKakuroAndAssign(String filepath) + fillBoardAux2(Cell[][] board, int i, int j, int[][] posComb, int[][][] + getKakuro(String filePath) + fillBoardAux(Cell[][] board, int i, int j, int[][] posComb, int[][][] t + saveKakuro() + fillBoard(Cell[][] board) + checkCoord(int x, int y) + howManyWhites(Cell[][] board) + generate(int size,int dif) + firstColRow(Cell[][] board,int dif) + randomCells(Cell[][] board,int dif) **CtrlResolve** + checkBoard(Cell[][] board) - kakuro: KAkuro + DFS(Cell[][] board,int i,int j,int[][] auxBoard) + CtrlResolve() + connexBoard(Cell[][] board) + setKakuro(Kakuro currentKakuro) + resolve(int r, int c, int sum, int [] vec) CtrlValidate CtrlPlay - kakuro: Kakuro - kakuro: Kakuro + CtrlValidate() + CtrlPlay() + setKakuro(Kakuro currentKakuro) + startGame(Kakuro newKakuro) + computePosSumsRec(int arr[], int index, int reducedNum, int lim, int posSums[], int no) + helpMyValue(int x, int y) + computePosSums(int x, int n, int no) + helpCorrectNumber(int x, int y) + validatePosSums(int [][][] tempBoard, int [] posSums, int num, int row, int i, int j) + isUnique(int [] a) + checkForNewUniques(int [][][] tempBoard) + setDifficulty() + howManyNumbers(int [] a) + validate(int r, int c, int sum, int [] vec, int [] res)

# **Presentation Controllers**

### CtrlUI

- cd: CtrlDomain
- Scanner = Scanner
- + CtrlUI()
- + run()
- + menu()
- + play()
- + help()
- + checkCoor(int x, int y)
- + readNumber()
- + readLine()
- + writeKakuroInTerminal()
- + writeCorrectKakuroInTerminal()

# **Persistance Controllers**

#### CtrlData

- ctrlData: CtrlData
- + getInstance()
- + CtrlData()
- + searchKakuro (int difficulty, int kakuroSizeRow, int kakuroSizeColumn)
- + getKakuro(String filePath)
- + saveKakuro(String content, int diff, int sizeRow, int sizeColumn)
- + saveKakuro(String content, int diff, int sizeRow, int sizeColumn)
- + getNumberOfFiles(int diff, int sizeRow, int sizeColumn)