**classBoard**

**attributes**

private static final int CELLS

private static final int PILES

private static Boolean debug

private Boolean winnable

private int movNum

private StdDeck d

private FreeCell[]

freeAry

private HomeCell[] pileAry

**methods**

public Board()

public void init()

private void fillRow()

public Boolean makeMove()

public void updateBoardStats()

private Boolean intoFreeCell()

**CLI**

public String toString()

public String buildCellsCLI()

private String buildRowCLI()

**Mapping**

StdCard sourceSwitch()

private int destSwitch()

**Utilities**

Int maxPileSize()

Public static void toggleDebug()

Public static void unitTest()

**package client gui**

attributes

public interface GUIInterface

private static fina long serialVersionUID

private static final String SEP

private static final String IMAGASDIR

private static final String CARDIMAGESDIR

private static Boolean debug

methods

public FreeGUI()

public static void Paint()

public static void consoleOut()

private void initComponents()

**package client**

class Driver

methods

public static void main()

**package board**

methods

boolean placeCard()

StdCard removeCard()

StdCard peekCard()

**package board FreeCell**

attributes

private static boolean debug

private StdCard cell

methods

public boolean placeCard()

public StdCard removeCard()

public StdCard peekCard()

private boolean isEmpty()

utilities

public String toString()

public static void toggleDebug()

public static void unitTest()

**package board HomeCell**

attributes

private static boolean debug

private StdCard cell

methods

public boolean placeCard()

public StdCard removeCard()

public stdCard oeekCard

public boolean isEmpty()

private boolean isSameSuit()

private boolean isAce

private boolean isValidCard()

utilities

public String toString()

public static void toggleDebug()

public static void unitTest()

**package board playing pile**

class PlayingPile

attributes

private static boolean debug

private ArrayList<StdCard> pile

methods

public PlayingPile()

void placeCardOnDeal()

boolean placeCard()

StdCard removeCard()

StdCard getCardAt()

StdCard peekLastCard()

int size()

private boolean isEmpty()

private boolean isDsc()

private boolean isAltColor()

private boolean isValid()

utilities

public String toString()

public static void toggleDebug()

public static void unitTest()

**package engine**

class Engine

attributes

private static boolean gui

private static boolean gameOver

private static boolean debug

private static Stack<Board> history

private static Board curBoard

methods

public static void start()

private static void gameLoop

private static void newDeal()

private static void undo()

private static void redo()

private static void hint()

utilities

public static void snapshot()

public static void printSnapshot()

public static void toggleDebug()

**package engine turn**

*class Turn*

attributes

private boolean winnable

private int movNum

private int String turnString

private Board board

methods

public Turn()

public boolean getWinnable()

public int getMoveNum()

private boolean isWinnable()

public void cliturn()

public void guiTurn()

utilities

public String toString()

private static String checkInput()

public static void toggleDebug()

public static void uniTest()

**package cli**

*class CLI*

attributes

private static boolean debug

methods

public static void prompt()

scan.close()

public static void game()

private static void toggleDebug()

private static void printHelp()

private static void formFeed()

private static void credz()

**package cli**

*class Debugger*

attributes

private static boolean debug

methods

public static void masterToggleDebug

**package cli**

*class DebugUnitTestTemplate*

attributes

private static boolean debug

methods

public static void toggleDebug()

public static void uniTes()

public void callUsingThese()

**package cli**

*class Tester*

methods

static void enter()

private static void testAll()

private static void customTestExample()

**package playingCards**

public interface CardInterface

methods

public getRank()

public getSuit()

**package playingCards**

public class DeckException extends java.lan.RuntimeException

method

public DeckException()

**package playingCards**

public interface DeckInterface

methods

public void shuffle()

public StdCard getCard()

public void print()

**package playingCards**

*public class StdCard implements CardInterface*

attributes

static final String[] rankAry

static final String[] suitAry

static final Character[] defSymAry

static final Character[] uniSymAry

private static boolean debug

private static boolean unicode

private int rank

private int suit

methods

public StdCard()

public StCard()

private void setRank()

private void setSuit()

public void toggleUnicode()

public static void toggleDebug()

public static void toggleDebug()

public int getRank()

public int getValue()

public int getSuit()

public String getRankString()

public String getSuitString()

public String getDefSymAry()

public String getUniSym()

public String getName()

public boolean isUnicode()

public booleasn isBlack()

testing

public static void unitTest()

**package playingCards**

public class StdDeck implements DeckInterface

*attributes*

private static final int SUITS

private static final int RANKS

private static final int SIZE

private static final Sring ERR1

private static intDeckCount

private static boolean debug

private StdCard

methods

public StdDeck()

public static void toggleDebug()

public int size()

public int getDeckCount()

public StdCard getCard() throws DeckException

public boolean isEmpty()

private void int()

public void shuffle()

public void print()

public void toggle Unicode()

public static void unitTest()

**package utils**

*class SysUtils*

attributes

private static final String SEP

private static final String PATH

methods

public static boolean isWindows()

public static String getPath()

public static String getSeparator()

public static void exitDoor()