

# Class Game

java.lang.Object  
Game

```
public class Game
extends java.lang.Object
```

## Constructor Summary

### Constructors

#### Constructor and Description

`Game ()`

## Method Summary

All MethodsStatic MethodsConcrete Methods

Modifier and Type	Method and Description
static java.lang.String	<b>canMove</b> (Board board, java.util.ArrayList<Piece> chessmanList, java.util.ArrayList<Piece> chessmanListEnemy, int arrPos, java.lang.String fx) Método que valida o moviento das peças de acordo com seu tipo.
static boolean	<b>isBoard</b> (java.lang.String boardPos) Método de validação que retorna se a posição (Letra,Número) faz parte do tabuleiro retorna true se for parte do tabuleiro.
static boolean	<b>isBoardAux</b> (java.lang.String boardPos, char l, int i) Método de auxiliar de validação que indica se a posição faz parte das instâncias do tabuleiro
static java.lang.String	<b>loadGame</b> (Board board, java.util.ArrayList<Piece> chessman1, java.util.ArrayList<Piece> chessman2, java.lang.String slot, int compatibility) Método que converte o char posição X em inteiro Parâmetros: char x cordenada no formato de número;
static void	<b>newGame</b> (Board board, java.util.ArrayList<Piece> chessman1, java.util.ArrayList<Piece> chessman2, int compatibility) Método que converte o char posição X em inteiro Parâmetros: char x cordenada no formato de número;
static boolean	<b>pawnChange</b> (Board board, Piece chessman, java.lang.String fx)
static int	<b>positionX</b> (char x) Método que converte o char posição X em inteiro Parâmetros: char x cordenada no formato de número;
static int	<b>positionY</b> (char y) Método que converte o char posição Y em inteiro Parâmetros: char y cordenada no formato de letra;

## Methods inherited from class java.lang.Object

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Constructor Detail

### Game

```
public Game()
```

## Method Detail

### canMove

```
public static java.lang.String canMove(Board board,
                                       java.util.ArrayList<Piece> chessmanList,
                                       java.util.ArrayList<Piece> chessmanListEnemy,
                                       int arrPos,
                                       java.lang.String fx)
```

Método que valida o movimento das peças de acordo com seu tipo. Parâmetros: Board board (Faz verificação de posição em relação ao tabuleiro) ArrayList chessmanList (Utilizada para getType da peça movida e para verificar se está se movendo para a posição de um aliado ou inimigo (caso board.isEmpty==false)) ArrayList chessmanListEnemy (Utilizado para validar os movimentos do rei, tendo em vista que o mesmo não pode se por em xeque) int arrPos (Posição da peça que será movida em relação ao seu ArrayList) String fx (Posição destino para o qual se valida o movimento) Retorno composto de comando e identificação de proibição; retorna String.charAt(0) == T se for possível se mover para a posição retorna String.charAt(0) == F se não for possível se mover para a posição retorna String.charAt(1) == C caso a peça possa se mover e esteja (ou não) apenas seja impedida por ser a posição de um aliado retorna String.charAt(1) == N caso a peça não possa se mover mesmo se a posição estiver vazia retorna String.charAt(1) == K caso a peça seja movida para a mesma posição de um inimigo (Prever remoção da peça que foi eliminada). retorna String.substring(2) == identificação da proibição.

### isBoard

```
public static boolean isBoard(java.lang.String boardPos)
```

Método de validação que retorna se a posição (Letra,Número) faz parte do tabuleiro retorna true se for parte do tabuleiro.

### isBoardAux

```
public static boolean isBoardAux(java.lang.String boardPos,
                                 char l,
                                 int i)
```

Método de auxiliar de validação que indica se a posição faz parte das instâncias do tabuleiro

### loadGame

```
public static java.lang.String loadGame(Board board,
                                       java.util.ArrayList<Piece> chessman1,
                                       java.util.ArrayList<Piece> chessman2,
```

```
        java.lang.String slot,  
        int compatibility)  
    throws java.io.FileNotFoundException
```

Método que converte o char posição X em inteiro Parâmetros: char x cordenada no formato de número;

**Throws:**

java.io.FileNotFoundException

**newGame**

```
public static void newGame(Board board,  
                           java.util.ArrayList<Piece> chessman1,  
                           java.util.ArrayList<Piece> chessman2,  
                           int compatibility)
```

Método que converte o char posição X em inteiro Parâmetros: char x cordenada no formato de número;

**pawnChange**

```
public static boolean pawnChange(Board board,  
                                 Piece chessman,  
                                 java.lang.String fx)
```

**positionX**

```
public static int positionX(char x)
```

Método que converte o char posição X em inteiro Parâmetros: char x cordenada no formato de número;

**positionY**

```
public static int positionY(char y)
```

Método que converte o char posição Y em inteiro Parâmetros: char y cordenada no formato de letra;