

Class **JavaCraft**

java.lang.Object
JavaCraft

public class **JavaCraft**
extends **Object**

Main game class.

Field Summary

Fields		
Modifier and Type	Field	Description
private static final int	AIR	The Integer value of AIR
private static final String	ANSI_BLUE	The ANSI color code for BLUE
private static final String	ANSI_BROWN	The ANSI color code for BROWN
private static final String	ANSI_CYAN	The ANSI color code for CYAN
private static final String	ANSI_GRAY	The ANSI color code for GRAY
private static final String	ANSI_GREEN	The ANSI color code for GREEN
private static final String	ANSI_PURPLE	The ANSI color code for PURPLE
private static final String	ANSI_RED	The ANSI color code for RED
private static final String	ANSI_RESET	The ANSI color code for RESET
private static final String	ANSI_WHITE	The ANSI color code for WHITE
private static final String	ANSI_YELLOW	The ANSI color code for YELLOW
private static final String	BLOCK_NUMBERS_INFO	The info on block numbers
private static final int	CRAFT_IRON_INGOT	The Integer value of CRAFT_IRON_INGOT
private static final int	CRAFT_STICK	The Integer value of CRAFT_STICK
private static final int	CRAFT_WOODEN_PLANKS	The Integer value of CRAFT_WOODEN_PLANKS
private static final int	CRAFTED_IRON_INGOT	The Integer value of CRAFTED_IRON_INGOT
private static final int	CRAFTED_STICK	The Integer value of CRAFTED_STICK
private static final int	CRAFTED_WOODEN_PLANKS	The Integer value of CRAFTED_WOODEN_PLANKS
private static List <Integer >	craftedItems	The players crafted items
private static final int	EMPTY_BLOCK	The Integer value of EMPTY_BLOCK
private static boolean	inSecretArea	The value of if the player is in the secret area.
private static List <Integer >	inventory	The players inventory
private static final int	INVENTORY_SIZE	The size of the inventory
private static final int	IRON_ORE	The Integer value of IRON_ORE
private static final int	LEAVES	The Integer value of LEAVES
private static int	NEW_WORLD_HEIGHT	The new world height
private static int	NEW_WORLD_WIDTH	The new world width
private static int	playerX	The players X position
private static int	playerY	The players Y position
private static Scanner	scanner	The Scanner to read input
private static boolean	secretDoorUnlocked	The value of if the secret door is unlocked.
private static final int	STONE	The Integer value of STONE
private static boolean	unlockMode	The value of the unlock mode.
private static final int	WOOD	The Integer value of WOOD
private static int[][]	world	The game world
private static int	worldHeight	The game worlds height
private static int	worldWidth	The game worlds width

Constructor Summary

Constructors	
Constructor	Description
JavaCraft()	

Method Summary

All Methods	Static Methods	Concrete Methods
Modifier and Type	Method	Description
static void	addCraftedItem(int craftedItem)	Adds a crafted item to craftedItems.

private static void	<code>clearScreen()</code>	Clears the screen.
static void	<code>craftIronIngot()</code>	Crafts CRAFTED_IRON_INGOT.
static void	<code>craftItem(int recipe)</code>	Crafts an item.
static void	<code>craftStick()</code>	Crafts CRAFTED_STICK.
static void	<code>craftWoodenPlanks()</code>	Crafts CRAFTED_WOODEN_PLANKS.
static void	<code>displayCraftingRecipes()</code>	Prints crafting recipes.
static void	<code>displayInventory()</code>	Prints players inventory.
static void	<code>displayLegend()</code>	Prints a legend.
static void	<code>displayWorld()</code>	Prints the world as ASCII text.
private static void	<code>fillInventory()</code>	Fills players inventory with all items.
private static void	<code>generateEmptyWorld()</code>	Generates an empty world.
static void	<code>generateWorld()</code>	Generates the world.
private static char	<code>getBlockChar(int blockType)</code>	Returns the symbol for blockType.
private static String	<code>getBlockColor(int blockType)</code>	Returns block color.
private static String	<code>getBlockName(int blockType)</code>	Returns human readable block name.
private static String	<code>getBlockSymbol(int blockType)</code>	Returns the symbol and color for blockType.
private static int	<code>getBlockTypeFromCraftedItem(int craftedItem)</code>	Returns the block type of craftedItem.
static void	<code>getCountryAndQuoteFromServer()</code>	Gets country and quote from server.
private static String	<code>getCraftedItemColor(int craftedItem)</code>	Returns item color.
private static int	<code>getCraftedItemFromBlockType(int blockType)</code>	Returns the crafted item of blockType.
private static String	<code>getCraftedItemName(int craftedItem)</code>	Returns human readable item name.
static void	<code>initGame(int worldWidth, int worldHeight)</code>	Initializes the game.
static void	<code>interactWithWorld()</code>	Handles interaction with the game world.
static boolean	<code>inventoryContains(int item)</code>	Queries inventory for an item.
static boolean	<code>inventoryContains(int item, int count)</code>	Queries inventory for if it has enough of an item.
static void	<code>loadGame(String fileName)</code>	Loads the game.
private static void	<code>lookAround()</code>	Prints all blocks surrounding the player.
static void	<code>main(String [] args)</code>	Main method.
static void	<code>mineBlock()</code>	Mines a block.
static void	<code>movePlayer(String direction)</code>	Moves the player
static void	<code>placeBlock(int blockType)</code>	Places a block.
static void	<code>removeItemsFromInventory(int item, int count)</code>	Removes a count of item from inventory.
private static void	<code>resetWorld()</code>	Resets the world to an empty world.
static void	<code>saveGame(String fileName)</code>	Saves the game.
static void	<code>startGame()</code>	Starts the game.
private static void	<code>waitForEnter()</code>	Waits for input ENTER.

Methods inherited from class java.lang.Object
<code>clone</code> , <code>equals</code> , <code>finalize</code> , <code>getClass</code> , <code>hashCode</code> , <code>notify</code> , <code>notifyAll</code> , <code>toString</code> , <code>wait</code> , <code>wait</code> , <code>wait</code>

Field Details

inSecretArea
<pre>private static boolean inSecretArea</pre>
The value of if the player is in the secret area.
Returns true if player is in the secret area, false in any other case.
Part of secret door logic.
secretDoorUnlocked
<pre>private static boolean secretDoorUnlocked</pre>
The value of if the secret door is unlocked.
Returns true if the secret door is unlocked, false in any other case.
Part of secret door logic.
unlockMode
<pre>private static boolean unlockMode</pre>
The value of the unlock mode.
Returns true if FIXME: TO ADD, false in any other case.
Part of secret door logic.

AIR

private static final int AIR

The Integer value of AIR

See Also:

[Constant Field Values](#)

CRAFT_IRON_INGOT

private static final int CRAFT_IRON_INGOT

The Integer value of CRAFT_IRON_INGOT

See Also:

[Constant Field Values](#)

CRAFT_STICK

private static final int CRAFT_STICK

The Integer value of CRAFT_STICK

See Also:

[Constant Field Values](#)

CRAFT_WOODEN_PLANKS

private static final int CRAFT_WOODEN_PLANKS

The Integer value of CRAFT_WOODEN_PLANKS

See Also:

[Constant Field Values](#)

CRAFTED_IRON_INGOT

private static final int CRAFTED_IRON_INGOT

The Integer value of CRAFTED_IRON_INGOT

See Also:

[Constant Field Values](#)

CRAFTED_STICK

private static final int CRAFTED_STICK

The Integer value of CRAFTED_STICK

See Also:

[Constant Field Values](#)

CRAFTED_WOODEN_PLANKS

private static final int CRAFTED_WOODEN_PLANKS

The Integer value of CRAFTED_WOODEN_PLANKS

See Also:

[Constant Field Values](#)

EMPTY_BLOCK

private static final int EMPTY_BLOCK

The Integer value of EMPTY_BLOCK

See Also:

[Constant Field Values](#)

INVENTORY_SIZE

private static final int INVENTORY_SIZE

The size of the inventory

See Also:

[Constant Field Values](#)

IRON_ORE

private static final int IRON_ORE

The Integer value of IRON_ORE

See Also:

[Constant Field Values](#)

LEAVES

private static final int LEAVES

The Integer value of LEAVES

See Also:

[Constant Field Values](#)

STONE

```
private static final int STONE
```

The Integer value of STONE

See Also:

[Constant Field Values](#)

WOOD

```
private static final int WOOD
```

The Integer value of WOOD

See Also:

[Constant Field Values](#)

ANSI_BLUE

```
private static final String ANSI_BLUE
```

The ANSI color code for BLUE

See Also:

[Constant Field Values](#)

ANSI_BROWN

```
private static final String ANSI_BROWN
```

The ANSI color code for BROWN

See Also:

[Constant Field Values](#)

ANSI_CYAN

```
private static final String ANSI_CYAN
```

The ANSI color code for CYAN

See Also:

[Constant Field Values](#)

ANSI_GRAY

```
private static final String ANSI_GRAY
```

The ANSI color code for GRAY

See Also:

[Constant Field Values](#)

ANSI_GREEN

```
private static final String ANSI_GREEN
```

The ANSI color code for GREEN

See Also:

[Constant Field Values](#)

ANSI_PURPLE

```
private static final String ANSI_PURPLE
```

The ANSI color code for PURPLE

See Also:

[Constant Field Values](#)

ANSI_RED

```
private static final String ANSI_RED
```

The ANSI color code for RED

See Also:

[Constant Field Values](#)

ANSI_RESET

```
private static final String ANSI_RESET
```

The ANSI color code for RESET

See Also:

[Constant Field Values](#)

ANSI_WHITE

```
private static final String ANSI_WHITE
```

The ANSI color code for WHITE

See Also:

[Constant Field Values](#)

ANSI_YELLOW

```
private static final String ANSI_YELLOW
```

The ANSI color code for YELLOW

See Also:

[Constant Field Values](#)

BLOCK_NUMBERS_INFO

```
private static final String BLOCK_NUMBERS_INFO
```

The info on block numbers

See Also:

[Constant Field Values](#)

NEW_WORLD_HEIGHT

```
private static int NEW_WORLD_HEIGHT
```

The new world height

NEW_WORLD_WIDTH

```
private static int NEW_WORLD_WIDTH
```

The new world width

playerX

```
private static int playerX
```

The players X position

playerY

```
private static int playerY
```

The players Y position

worldHeight

```
private static int worldHeight
```

The game worlds height

worldWidth

```
private static int worldWidth
```

The game worlds width

world

```
private static int[][] world
```

The game world

craftedItems

```
private static List <Integer> craftedItems
```

The players crafted items

inventory

```
private static List <Integer> inventory
```

The players inventory

scanner

```
private static Scanner scanner
```

The Scanner to read input

Constructor Details

JavaCraft

```
public JavaCraft()
```

Method Details

main

```
public static void main(String [] args)
```

Main method.

This method is called upon execution of the game.

Parameters:

args - The supplied commandline arguments

initGame

```
public static void initGame(int worldWidth,
                           int worldHeight)
```

Initializes the game.

This method sets `JavaCraft.worldWidth`, `JavaCraft.worldHeight`, `JavaCraft.world`, `playerX`, `playerY` and initializes inventory.

Parameters:

`worldWidth` - The width of world in blocks

`worldHeight` - The height of world in blocks

generateWorld

```
public static void generateWorld()
```

Generates the world.

This method uses randomness to generate a world out of different materials.

displayWorld

```
public static void displayWorld()
```

Prints the world as ASCII text.

This method is responsible for displaying the world.

getBlockSymbol

```
private static String getBlockSymbol(int blockType)
```

Returns the symbol and color for `blockType`.

This method returns the mapped char and `blockColor` for `blockType`.

Parameters:

`blockType` - The type of block

Returns:

String The mapped symbol and `blockColor` for `blockType`

getBlockChar

```
private static char getBlockChar(int blockType)
```

Returns the symbol for `blockType`.

This method returns the mapped char for `blockType`.

Parameters:

`blockType` - The type of block

Returns:

char The mapped symbol for `blockType`

startGame

```
public static void startGame()
```

Starts the game.

This method handles the following:

- Printing of initial UI, instructions and informational messages
- Player input
- Secret door logic

Part of secret door logic.

fillInventory

```
private static void fillInventory()
```

Fills players inventory with all items.

This method fills the players inventory with all available `blockTypes`.

Part of secret door logic.

resetWorld

```
private static void resetWorld()
```

Resets the world to an empty world.

This method resets the world to an empty world via generating an empty world and resetting the players position.

Part of secret door logic.

generateEmptyWorld

```
private static void generateEmptyWorld()
```

Generates an empty world.

This method generates an empty world which is part of the secret door logic.

Part of secret door logic.

clearScreen

```
private static void clearScreen()
```

Clears the screen.

This method clears the screen and uses different logic depending on the OS.

Catched Exceptions:

- On IOException: Prints stacktrace when I/O exception of some sort has occurred.
- On InterruptedException: Prints stacktrace when a thread is waiting, sleeping, or otherwise occupied, and the thread is interrupted, either before or during the activity.

lookAround

```
private static void lookAround()
```

Prints all blocks surrounding the player.

This method prints all blocks surrounding the player. This is meant to make the players life easier.

movePlayer

```
public static void movePlayer(String direction)
```

Moves the player

This method moves the player UP/DOWN/LEFT/RIGHT depending on the supplied direction.

Parameters:

direction - The direction the player should be moved towards.

mineBlock

```
public static void mineBlock()
```

Mines a block.

This method mines a block and adds it to the players inventory if it is not AIR.

placeBlock

```
public static void placeBlock(int blockType)
```

Places a block.

This method places a block that is of blockType 0 to 7 and removes it from the players inventory if the players inventory contains that block.

Parameters:

blockType - The type of block

getBlockTypeFromCraftedItem

```
private static int getBlockTypeFromCraftedItem(int craftedItem)
```

Returns the block type of craftedItem.

This method returns the block type of craftedItem.

Defaults to -1.

Parameters:

craftedItem - The crafted item

Returns:

int The block type of craftedItem

getCraftedItemFromBlockType

```
private static int getCraftedItemFromBlockType(int blockType)
```

Returns the crafted item of blockType.

This method returns the crafted item of blockType.

Defaults to -1.

Parameters:

blockType - The type of block

Returns:

int The crafted item of blockType

displayCraftingRecipes

```
public static void displayCraftingRecipes()
```

Prints crafting recipes.

This method prints the available crafting recipes.

craftItem

```
public static void craftItem(int recipe)
```

Crafts an item.

This method crafts an item from a recipe.

Prints message if invalid recipe was supplied.

Parameters:

`recipe` - The recipe used to craft the item

craftWoodenPlanks

```
public static void craftWoodenPlanks()
```

Crafts CRAFTED_WOODEN_PLANKS.

This method crafts CRAFTED_WOODEN_PLANKS from 2 WOOD that are taken from the players inventory.

Prints message if the player doesn't have the correct items in his inventory.

craftStick

```
public static void craftStick()
```

Crafts CRAFTED_STICK.

This method crafts CRAFTED_STICK from 1 WOOD that is taken from the players inventory.

Prints message if the player doesn't have the correct items in his inventory.

craftIronIngot

```
public static void craftIronIngot()
```

Crafts CRAFTED_IRON_INGOT.

This method crafts CRAFTED_IRON_INGOT from 3 IRON_ORE that is taken from the players inventory.

Prints message if the player doesn't have the correct items in his inventory.

inventoryContains

```
public static boolean inventoryContains(int item)
```

Queries inventory for an item.

This method queries the players inventory for an item.

Parameters:

`item` - The item to query the inventory for

Returns:

boolean true if inventory contains item, false in any other case

inventoryContains

```
public static boolean inventoryContains(int item,
                                       int count)
```

Queries inventory for if it has enough of an item.

This method queries the players inventory for an item and if it contains at least as much as the supplied count.

Parameters:

`item` - The item to query the inventory for

`count` - The count that the inventory should contain of the item

Returns:

boolean true if inventory contains item at least as many times as the supplied count, false in any other case

removeItemsFromInventory

```
public static void removeItemsFromInventory(int item,
                                           int count)
```

Removes a count of item from inventory.

This method removes a count of an item from the players inventory.

Parameters:

`item` - The item to remove from the inventory

`count` - The count that should be removed from the inventory

addCraftedItem

```
public static void addCraftedItem(int craftedItem)
```

Adds a crafted item to craftedItems.

This method adds a crafted item to craftedItems that are part of the players inventory.

Parameters:

`craftedItem` - The crafted item

interactWithWorld

```
public static void interactWithWorld()
```

Handles interaction with the game world.

This method handles interaction with the game world and prints messages for blocks that the player can interact with. It also adds certain blocks to the players inventory if he interacts with them.

saveGame

```
public static void saveGame(String fileName)
```

Saves the game.

This method saves the game in a file.

Parameters:

fileName - The file name

Catched Exceptions:

- On IOException: Prints error with message when I/O exception of some sort has occurred.

loadGame

```
public static void loadGame(String fileName)
```

Loads the game.

This method loads the game from a file.

Parameters:

fileName - The file name

Catched Exceptions:

- On IOException: Prints error with message when I/O exception of some sort has occurred.
- On ClassNotFoundException: Prints error with message when no definition for the class with the specified name could be found.

getBlockName

```
private static String getBlockName(int blockType)
```

Returns human readable block name.

This method returns a human readable block name for blockType.

Defaults to "Unknown"

Parameters:

blockType - The type of block

Returns:

String The human readable block name.

displayLegend

```
public static void displayLegend()
```

Prints a legend.

This method prints a legend of items on the map.

displayInventory

```
public static void displayInventory()
```

Prints players inventory.

This method prints the players inventory including craftedItems.

getBlockColor

```
private static String getBlockColor(int blockType)
```

Returns block color.

This method returns the blocks color.

Defaults to empty String

Parameters:

blockType - The type of block

Returns:

String The human readable name of craftedItem

waitForEnter

```
private static void waitForEnter()
```

Waits for input ENTER.

This method waits for player to input ENTER.

getCraftedItemName

```
private static String getCraftedItemName(int craftedItem)
```

Returns human readable item name.

This method returns a human readable item name for craftedItem.

Parameters:

craftedItem - The crafted item

Returns:

String The human readable name of craftedItem

getCraftedItemColor

```
private static String getCraftedItemColor(int craftedItem)
```

Returns item color.

This method returns the items color.

Defaults to empty String

Parameters:

craftedItem - The crafted item

Returns:

String The human readable name of craftedItem

getCountryAndQuoteFromServer

```
public static void getCountryAndQuoteFromServer()
```

Gets country and quote from server.

This method gets country and quote from server via a POST request.

Catched Exceptions:

- On Exception: Prints an error for any encountered exception.