## Class JavaCraft

    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_COAL_GRAY** | The ANSI color code for GRAY |
| private static final **String** | **ANSI_CYAN** | The ANSI color code for CYAN |
| private static final **String** | **ANSI_EMERALD_GREEN** | The ANSI color code for EMERALD_GREEN |
| 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 | **COAL_ORE** | The Integer value of COAL_ORE |
| private static final int | **CRAFT_IRON_INGOT** | The Integer value of CRAFT_IRON_INGOT |
| private static final int | **CRAFT_IRON_PICKAXE** | The Integer value of CRAFT_IRON_PICKAXE |
| private static final int | **CRAFT_STICK** | The Integer value of CRAFT_STICK |
| private static final int | **CRAFT_STONE_PICKAXE** | The Integer value of CRAFT_STON_PICKAXE |
| 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_IRON_PICKAXE** | The Integer value of CRAFTED_IRON_PICKAXE |
| private static final int | **CRAFTED_STICK** | The Integer value of CRAFTED_STICK |
| private static final int | **CRAFTED_STONE_PICKAXE** | The Integer value of CRAFTED_STONE_PICKAXE |
| 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 | **EMERALD_ORE** | The Integer value of EMERALD_ORE |
| 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 | 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 | **clearScreen**() | Clears the screen. |
| static boolean | **craftedItemsContains**(int craftedItem) | Queries craftedItems for an item. |
| static boolean | **craftedItemsContains**(int craftedItem, int count) | Queries craftedItems for if it has enough of an crafted item. |
| static void | **craftIronIngot**() | Crafts CRAFTED_IRON_INGOT. |
| static void | **craftIronPickaxe**() | Crafts CRAFTED_IRON_PICKAXE. |
| static void | **craftItem**(int recipe) | Crafts an item. |
| static void | **craftStick**() | Crafts CRAFTED_STICK. |
| static void | **craftStonePickaxe**() | Crafts CRAFTED_STONE_PICKAXE. |
| static void | **craftWoodenPlanks**() | Crafts CRAFTED_WOODEN_PLANKS. |
| static void | **displayCraftingRecipes**() | Prints crafting recipes. |
| static void | **displayInventory**() | Prints players inventory. |
| static void | **displayLegend**() | Prints a legend. |
| static void | **displayWorld**() | Prints the world as ASCII text. |
| private static void | **fillInventory**() | Fills players inventory with all blocks. |
| private static void | **generateEmptyWorld**() | Generates an empty world. |
| static void | **generateWorld**() | Generates the world. |
| private static char | **getBlockChar**(int blockType) | Returns the symbol for blockType. |
| private static **String** | **getBlockColor**(int blockType) | Returns block color. |
| private static **String** | **getBlockName**(int blockType) | Returns human readable block name. |
| private static **String** | **getBlockSymbol**(int blockType) | Returns the symbol and color for blockType. |
| private static int | **getBlockTypeFromCraftedItem**(int craftedItem) | Returns the block type of craftedItem. |
| static void | **getCountryAndQuoteFromServer**() | Gets country and quote from server. |
| private static **String** | **getCraftedItemColor**(int craftedItem) | Returns item color. |
| private static int | **getCraftedItemFromBlockType**(int blockType) | Returns the crafted item of blockType. |
| private static **String** | **getCraftedItemName**(int craftedItem) | Returns human readble item name. |
| static int | **getMineRequFromBlockType**(int blockType) | Returns the crafted item that is required to mine blockType. |
| static void | **initGame**(int worldWidth, int worldHeight) | Initializes the game. |
| static void | **interactWithWorld**() | Handles interaction with the game world. |
| static boolean | **inventoryContains**(int item) | Queries inventory for an item. |
| static boolean | **inventoryContains**(int item, int count) | Queries inventory for if it has enough of an item. |
| static void | **loadGame**(**String** fileName) | Loads the game. |
| private static void | **lookAround**() | Prints all blocks sorrounding the player. |
| static void | **main**(**String** [] args) | Main method. |
| static void | **mineBlock**() | Mines a block. |
| static void | **movePlayer**(**String** direction) | Moves the player |
| static void | **placeBlock**(int blockType) | Places a block. |
| static void | **removeItemFromCraftedItems**(int craftedItem, int count) | Removes a count of item from craftedItem. |
| static void | **removeItemsFromInventory**(int item, int count) | Removes a count of item from inventory. |
| private static void | **resetWorld**() | Resets the world to an empty world. |
| static void | **saveGame**(**String** fileName) | Saves the game. |
| static void | **startGame**() | Starts the game. |
| private static void | **waitForEnter**() | Waits for input ENTER. |

### Methods inherited from class java.lang.**Object**

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

## Field Details

### inSecretArea

private static boolean inSecretArea

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

```
private static boolean secretDoorUnlocked
```

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

```
private static boolean unlockMode
```

The value of the unlock mode.

Returns true if unlock has been entered, 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_PICKAXE

```
private static final int CRAFT_IRON_PICKAXE
```

The Integer value of CRAFT_IRON_PICKAXE

**See Also:**
Constant Field Values

### CRAFT_STONE_PICKAXE

```
private static final int CRAFT_STONE_PICKAXE
```

The Integer value of CRAFT_STON_PICKAXE

**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_PICKAXE

```
private static final int CRAFTED_IRON_PICKAXE
```

The Integer value of CRAFTED_IRON_PICKAXE

**See Also:**
Constant Field Values

### CRAFTED_STONE_PICKAXE

```
private static final int CRAFTED_STONE_PICKAXE
```

The Integer value of CRAFTED_STONE_PICKAXE

**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

## EMERALD_ORE

```
private static final int EMERALD_ORE
```

The Integer value of EMERALD_ORE

**See Also:**

Constant Field Values

## COAL_ORE

```
private static final int COAL_ORE
```

The Integer value of COAL_ORE

**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:**

## ANSI_BROWN

```
private static final String ANSI_BROWN
```

The ANSI color code for BROWN

**See Also:**

## ANSI_CYAN

```
private static final String ANSI_CYAN
```

The ANSI color code for CYAN

**See Also:**

## ANSI_GRAY

```
private static final String ANSI_GRAY
```

The ANSI color code for GRAY

**See Also:**

## ANSI_GREEN

```
private static final String ANSI_GREEN
```

The ANSI color code for GREEN

**See Also:**

## ANSI_EMERALD_GREEN

```
private static final String ANSI_EMERALD_GREEN
```

The ANSI color code for EMERALD_GREEN

**See Also:**

## ANSI_COAL_GRAY

```
private static final String ANSI_COAL_GRAY
```

The ANSI color code for GRAY

**See Also:**

## ANSI_PURPLE

```
private static final String ANSI_PURPLE
```

The ANSI color code for PURPLE

**See Also:**

## ANSI_RED

```
private static final String ANSI_RED
```

The ANSI color code for RED

**See Also:**

## ANSI_RESET

```
private static final String ANSI_RESET
```

The ANSI color code for RESET

**See Also:**

## ANSI_WHITE

```
private static final String ANSI_WHITE
```

The ANSI color code for WHITE

**See Also:**

## 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 worldWidth, JworldHeight, 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.

Part of secret door logic.

## 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 blocks.

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 only contains the dutch flag.

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 sorrounding the player.

This method prints all blocks sorrounding 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 9 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

## getMineRequFromBlockType

```
public static int getMineRequFromBlockType(int blockType)
```

Returns the crafted item that is required to mine blockType.

This method returns the crafted item that is required to mine blockType.

Defaults -1.

**Parameters:**

`blockType` - The type of block

**Returns:**

int The crafted Item required to mine 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

## craftStonePickaxe

`public static void craftStonePickaxe()`

Crafts CRAFTED_STONE_PICKAXE.

This method crafts CRAFTED_STONE_PICKAXE from 1 Stick and 3 Stone that are taken form the players inventory.

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

## craftIronPickaxe

`public static void craftIronPickaxe()`

Crafts CRAFTED_IRON_PICKAXE.

This method crafts CRAFTED_IRON_PICKAXE from 1 Stick and 3 Iron Ingots that are taken form the players inventory.

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

## 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

---

### craftedItemsContains

```
public static boolean craftedItemsContains(int craftedItem)
```

Queries craftedItems for an item.

This method queries the players crafted item inventory for an item.

**Parameters:**

`craftedItem` - The item to query the crafted item inventory for

**Returns:**

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

---

### craftedItemsContains

```
public static boolean craftedItemsContains(int craftedItem,
                                            int count)
```

Queries craftedItems for if it has enough of an crafted item.

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

**Parameters:**

`craftedItem` - The crafted item to query the crafted items inventory for

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

**Returns:**

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

---

### removeItemFromCraftedItems

```
public static void removeItemFromCraftedItems(int craftedItem,
                                              int count)
```

Removes a count of item from craftedItem.

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

**Parameters:**

`craftedItem` - The item to remove from the crafted items inventory

`count` - The count that should be removed from the crafted items 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 readble 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.