| initgame | Sets the dimensions of the game world, creates a 2D array to represent the world, position the player in the center, and initializes an empty inventory list. |
| --- | --- |
| generateWorld | Generates the 2D array using random values between 0 and 100. Randomly assigns blocks as wood if the random number generated is less than 20, stone if the random number is less than 50, iron if the random number is less than 70, leaves if the random number is less than 35 and the rest of the world is filled with air. |
| displayWorld | Displays the world and outlines it with a border. If the player is in the secret area, the “P” will be blue, else it will be green. |
| getBlocksymbol | Returns a string representing that block type with associated ANSI color codes. |
| getBlockChar | Returns a character representing that block type. It uses a switch statement to specific block types (e.g., WOOD, LEAVES, STONE, IRON\_ORE) to corresponding Unicode characters (e.g., '\u2592', '\u00A7', '\u2593', '\u00B0'). |
| fillInventory | Ensures the inventory is empty and makes each block type have a maximum storage of 100 blocks. |
| resetWorld | Resets the world and positions the player in the center of the world. |
| generateEmptyWorld | Generates a new world and it fills in the top, middle and bottom stripes with white, red and blue blocks. |
| clearScreen | Checks if the player's operating system is Windows, if yes then it clears the terminal using the “cls” command. If not, it uses ANSI escape codes ‘’\033[H\033[2J’’ to clear the screen. |
| lookAround | Displays a 3x3 square with the player in the middle and the blocks that surround the player. |
| movePlaye | Moves the player in the 2D world ( x and y axis). |
| mineBlock | If the block is different from Air, it adds the block in the inventory and tells what you have mined. If the block is Air, it tells the user that there is nothing to be mined. |
| placeBlock | Waits for the user’s input to be a number between 0 (inclusive) and 7 ( inclusive) the number representing the block type if the input <= 4, else representing a crafted item. After the number was entered, it places the said block/Item on the player’s position. |
| getBlockTypeFromCraftedItem | Converts a block type to a crafted item. |
| getCraftedItemFromBlockType | Converts a crafted item to a block type. |
| displayCraftingRecipes | Displays all possible crafting recipes. |
| craftItem | Crafts an item based on the player’s input, 1 being for wooden planks, 2 for sticks and 3 for iron ingots. |
| craftWoodenPlanks | If there are at least 2 pieces of wood in inventory, remove the 2 pieces of wood from inventory and replace them with a wooden plank. |
| craftSticks | If there is at least 1 piece of wood in inventory, remove the 1 piece of wood from inventory and replace them with a stick. |
| craftIronIngots | If there are at least 3 pieces of iron ore in inventory, remove the 3 pieces of n from inventory and replace them with an iron ingot. |

| inventoryContains | Is a boolean function which checks if the player has enough blocks of the necessary block type so he can craft, if yes, it returns true, if not, it returns false. |
| --- | --- |
| removeItemsfromInventory | Removes the number of an item from inventory. This is used after crafting. |
| addCraftedItem | If the player didn’t have an item before, it creates a new ArrayList named craftedItems before adding the item in the inventory. |
| interactWithWorld | When the player input = “I”, checks if there is any block Type on the player’s coordinates. If there is, it adds the block which was interacted with in the inventory. |
| saveGame | Serialize game state data and write to the file. Saves data such as NEW\_WORLD\_WIDTH, NEW\_WORLD\_HEIGHT, world, playerX, playerY, inventory, craftedItems, unlockMode. Also tells you if the file was saved ( if yes, where?) or there was an error. |
| loadGame | Reads the file in which the world was saved. reads data such as NEW\_WORLD\_WIDTH, NEW\_WORLD\_HEIGHT, world, playerX, playerY, inventory, craftedItems, unlockMode. Also tells you if the file was loaded( if yes, from where?) or there was an error. |
| getBLockname | Takes the parameter blockType and returns a name for each block. |
| displayLegend | Displays the legend of the World, shows the characters and color that are assigned to each block. |
| displayInventory | Displays the player’s inventory and the count of items |
| getBlockColor | Gives the blocks color using the ANSI escape code method, using a switch statement, the case defines the block type and gives the said block its color. |
| waitForEnter | Waits for the user’s “Enter” input so the code can continue |
| getCraftedItemName | Takes the parameter craftedItem and returns a name for each crafted item. |
| getCrafteditemcolor | Takes the parameter craftedItem and assigns the color brown to all crafted  items. |
| getCountryanQuotefromServer | It establishes a connection to a remote server using HTTP POST. It sends an empty JSON payload to the server and reads the server's response as a JSON string. It extracts and prints a "country" and a "quote" from specific positions in the JSON response. |
| startGame | Starts the game, sets multiple boolean functions to false, prints a text showing are the player’s possible moves. It calls different existing functions in the code depending on what the user input is. |