

---

# Crafting these Days

---

MINECRAFT FOR THE LAZY!

VICTANA  
version 1.0.2  
August 4, 2015

# Contents

<b>1</b>	<b>World Gen</b>	<b>3</b>
<b>2</b>	<b>First Age</b>	<b>4</b>
2.1	Copper Wire . . . . .	4
2.2	Copper Cable . . . . .	4
2.3	Copper Energy Cable . . . . .	4
2.4	Soldering Iron . . . . .	5
2.5	Solder-wire . . . . .	5
2.6	Soldering Station . . . . .	5
2.7	Electrical Circuit . . . . .	6
2.8	ALU . . . . .	6
2.9	CPU . . . . .	6
2.10	Monitor . . . . .	7
2.11	Redstone Emitter . . . . .	7
2.12	Redstone Receiver . . . . .	7
2.13	Pipes . . . . .	8
2.14	Extraction Pipe . . . . .	8
2.15	Insertion Pipe . . . . .	9
2.16	Computer . . . . .	9
<b>3</b>	<b>Second Age</b>	<b>10</b>
3.1	Advanced Electronic Circuit . . . . .	10
3.2	Memory Chip . . . . .	10
3.3	Disk Driver . . . . .	10
3.4	Pulse Generator . . . . .	11
3.5	Mother Board . . . . .	11
3.6	Graphical Card . . . . .	11
3.7	Fan . . . . .	12
3.8	Desktop PC . . . . .	12
3.9	Monitor Tier 2 . . . . .	12
3.10	Advanced Copper Cable . . . . .	13
3.11	Filter Pipe . . . . .	13
3.12	SqDisc . . . . .	13
3.13	Hard Drive Port . . . . .	14
3.14	Local Server . . . . .	14
3.15	Hard Drive . . . . .	14
3.16	Shredder . . . . .	15
3.17	Printer . . . . .	15
3.18	Computer Tier 2 . . . . .	16
<b>4</b>	<b>Third Age</b>	<b>18</b>
4.1	Distronium Cube . . . . .	18
4.2	Enderium Circuit . . . . .	18
4.3	Enderium Cable . . . . .	18
4.4	Ender Receiver . . . . .	19
4.5	Ender Transmitter . . . . .	19

4.6	Wireless Modem . . . . .	19
4.7	Gate . . . . .	20
4.8	The Cube . . . . .	20
4.9	UDB port . . . . .	21
4.10	UDB . . . . .	21
4.11	Laptop . . . . .	21
<b>5</b>	<b>Tools and Utils</b>	<b>22</b>
5.1	Calculator . . . . .	22
5.2	Hammer . . . . .	22
5.3	Security Station . . . . .	23
<b>6</b>	<b>Code</b>	<b>24</b>
6.1	Basic Coding . . . . .	24
6.1.1	If statement . . . . .	24
6.1.2	While statement . . . . .	24
6.1.3	For statement . . . . .	24
6.1.4	Parameters and Variables . . . . .	24
6.2	Advanced Coding . . . . .	24
6.3	Commands . . . . .	24

## 1 World Gen

<b>Blocks</b>	<b>Drop</b>	<b>Tool Level</b>	<b>Ore rarity</b>
Copper	1 Copper ore	Stone	Coal
Tin	1 Tin ore	Stone	Coal
Distronium	1-3 Distronium Crystals	Diamond	Diamond

## 2 First Age

### 2.1 Copper Wire

#### Crafting:

Copper Ingot	Copper Ingot	Copper Ingot

**Makes:** 6 Copper Wires

Used as crafting component.

### 2.2 Copper Cable

#### Crafting:

Stone	Stone	Stone
Copper Wire	Stone	Copper Wire
Stone	Stone	Stone

**Makes:** 2 Copper Cable

Is used to connect machines together for the computer. Machines, Inventories and special blocks can also be linked by the use of electric circuits.

### 2.3 Copper Energy Cable

#### Crafting:

Copper Wire	Copper Wire	Copper Wire
Redstone Dust	Redstone Dust	Redstone Dust
Copper Wire	Copper Wire	Copper Wire

**Makes:** 3 Copper Energy Cable

Can carry up to 160 Rf/t.

## 2.4 Soldering Iron

### Crafting:

	Iron Ingot	
Iron Ingot		
Copper Wire		

Used to make electrical circuits in the Soldering Station.

## 2.5 Solder-wire

### Crafting:

	Tin Ingot	
Tin Ingot		Tin Ingot
	Tin Ingot	

**Makes:** 12 Solder-wire

Used as solder in the soldering station.

## 2.6 Soldering Station

### Crafting:

Iron Plate	Iron Plate	Iron Plate
Stone	Crafting Table	Stone
Stone	Stone	Stone

Used to make electrical circuits. It has 4 slots. 1 for coal, 1 for the soldering iron, 1 for Solder-wire and the 3x3 crafting grid. In the 3x3 crafting grid, electrical circuits will be made and then using solder and the soldering iron it will be bound together.

## 2.7 Electrical Circuit

### Crafting (Soldering Station):

Copper Wire	Redstone Dust	Copper Wire
Redstone Dust	Iron Plate	Redstone Dust
Copper Wire	Redstone Dust	Copper Wire

Basic Component for Electrical Items and Blocks. Can also be placed on the backs of e.g. Furnaces to connect them to the system.

## 2.8 ALU

### Crafting (Soldering Station):

Repeater	Copper Wire	Repeater
Redstone Dust	Comparator	Redstone Dust
Electrical Circuit	Redstone Dust	Electrical Circuit

Crafting component.

## 2.9 CPU

### Crafting:

Copper Wire	Electrical Circuit	Copper Wire
Piston	ALU	Repeater
Redstone Torch	Comparator	Clock

The CPU is part of the First Age Computer Multiblock. The CPU takes care of all the orders a task needs to complete. Each CPU increases the Data Memory by 16k bytes.

## 2.10 Monitor

### Crafting:

Glass Pane	Copper Wire	
Glass Pane	Electrical Circuit	Copper Wire
Glass Pane	Copper Wire	

Used to watch and program the local network it's connected to.

## 2.11 Redstone Emitter

### Crafting:

Stone	Electrical Circuit	Stone
Redstone Dust	Redstone Block	Redstone Dust
Stone	Redstone Dust	Stone

When connected to the network, a program can activate the Redstone Emitter to give a redstone strength of 15.

## 2.12 Redstone Receiver

### Crafting:

Stone	Electrical Circuit	Stone
Redstone Dust	Repeater	Redstone Dust
Stone	Redstone Dust	Stone

Can be checked if it receives a redstone signal.

## 2.13 Pipes

### Wooden Gear:

#### Crafting:

	Stick	
Stick		Stick
	Stick	

Crafting Component.

### Basic Pipe

#### Crafting:

Wooden Planks	Wooden Gear	Wooden Planks
Wooden Planks	Wooden Gear	Wooden Planks

Can be used to transport items.

## 2.14 Extraction Pipe

#### Crafting:

Wooden Planks	Wooden Gear	Wooden Planks	Basic Pipe	Sticky Piston	
	Sticky Piston				
Wooden Planks	Wooden Gear	Wooden Planks			

On a redstone signal, the extraction pipe pulls items from the attached inventory.

## 2.15 Insertion Pipe

### Crafting:

Wooden Planks	Wooden Gear	Wooden Planks	Basic Pipe	Piston	
	Piston				
Wooden Planks	Wooden Gear	Wooden Planks			

On a redstone signal, the insertion pipe pushes items into the attached inventory.

## 2.16 Computer

The First Age Computer is somewhat of a multiblock structure. The Computer itself is basically the CPU, but you cannot code it without a monitor. So these are the minimum requirements to make functional computer. "Functional", but without any further attachments it can't do anything but some calculations.

Connect machines through electric circuits to connect them to the network and interact with them. The commands you can use can be found in chapter 6.3. [Click Here](#)

### 3 Second Age

#### 3.1 Advanced Electronic Circuit

**Crafting (Soldering Station):**

Copper Wire	Electrical Circuit	Copper Wire
Electrical Circuit	Redstone Block	Electrical Circuit
Copper Wire	Electrical Circuit	Copper Wire

Crafting Component. Can also be used on certain blocks to make them connect-able to the network. (Gives more features than Electrical Circuit.)

#### 3.2 Memory Chip

**Crafting (Soldering Station):**

Comparator	Chest	Comparator
Chest	Advanced Electronic Circuit	Chest
Comparator	Chest	Comparator

Crafting Component.

#### 3.3 Disk Driver

**Crafting (Soldering Station):**

Sticky Piston	Tin Plate	
Tin Plate	Advanced Electronic Circuit	Tin Plate
Stick Piston	Tin Plate	

Crafting Component.

### 3.4 Pulse Generator

**Crafting:**

Copper Wire	Electrical Circuit	Copper Wire
Glass	Diamond	Glass
Clock	Redstone Torch	Repeater

Crafting Component.

### 3.5 Mother Board

**Crafting (Soldering Station):**

Memory Chip	Piston	Advanced Electrical Circuit
Repeater	CPU	Redstone Receiver
Pulse Generator	Advanced Electrical Circuit	Redstone Emitter

Crafting Component.

### 3.6 Graphical Card

**Crafting (Soldering Station):**

Copper Wire	Red Dye	Copper Wire
Advanced Electrical Circuit	Green Dye	Advanced Electrical Circuit
Copper Wire	Blue Dye	Copper Wire

Crafting Component.

### 3.7 Fan

#### Crafting:

	Iron Plate	
Iron Plate	Iron Ingot	Iron Plate
	Iron Plate	Copper Wire

Crafting Component.

### 3.8 Desktop PC

#### Crafting:

Iron Plate	Iron Plate	Disk Driver
Graphical Card	Mother Board	Iron Plate
Copper Wire	Iron Plate	Fan

A Compact and upgraded version of the CPU from the First Age.

### 3.9 Monitor Tier 2

#### Crafting:

Glass Pane	Copper Wire	
Glass Pane	Graphical Card	Advanced Electrical Circuit
Glass Pane	Copper Wire	

Better than the First Age Monitor, this monitor comes with color and a fancy GUI for easy input.

### 3.10 Advanced Copper Cable

#### Crafting:

Iron Plate	Copper Cable	Iron Plate
Insertion Pipe	Electrical Circuit	Extraction Pipe
Iron Plate	Repeater	Iron Plate

**Makes:** 8 Advanced Copper Cable

An advanced cable version that can only interact with the Advanced Electrical Circuit and can only be operated by a Second Age Computer Network. Through this Cables, Pushing, Pulling and sending Redstone signals is possible.

### 3.11 Filter Pipe

#### Crafting:

Piston	Red Dye	ink sac
Blue Dye	Advanced Copper Cable	Green Dye
white Dye	Yellow Dye	Piston

Can be programmed to set different routes for unset destinations of passing items. This is a way to push items from an advanced copper cable to a regular Basic Pipe.

### 3.12 SqDisc

**Crafting:** SqDisc, or Square Disc, is the minecraft version of a CDROM. You can put it in the

	Tin Plate	
Tin Plate		Tin Plate
	Tin Plate	

Desktop PC and read or write programs on it. These Discs can then be used as back-up or can be hand down to friends.

### 3.13 Hard Drive Port

**Crafting:**

		Tin Plate
Advanced Electronic Circuit	Sticky Piston	
		Tin Plate

Crafting Component.

### 3.14 Local Server

**Crafting:**

Hard Drive Port	Tin Plate	Hard Drive Port
Hard Drive Port	Filter Pipe	Hard Drive Port
Memory Chip	Advanced Electronic Circuit	Memory Chip

Can be used to store up to 4 Hard Drives.

### 3.15 Hard Drive

**Crafting:**

	Comparator	Advanced Electronic Circuit
Tin Plate	Sticky Piston	Memory Chip
	Redstone Repeater	Advanced Electronic Circuit

Can be used to store items from the shredder inside the Computer Network. From here items can also be pulled and sent to the printer.

### 3.16 Shredder

#### Crafting:

Piston		Piston
Tin Plate	Iron Sword	Tin Plate
Tin Plate	Advanced Electronic Circuit	Tin Plate

Throw in items to destroy them, convert them into digits and sent them into the network. Only Hard Drives can store this kind of data.

### 3.17 Printer

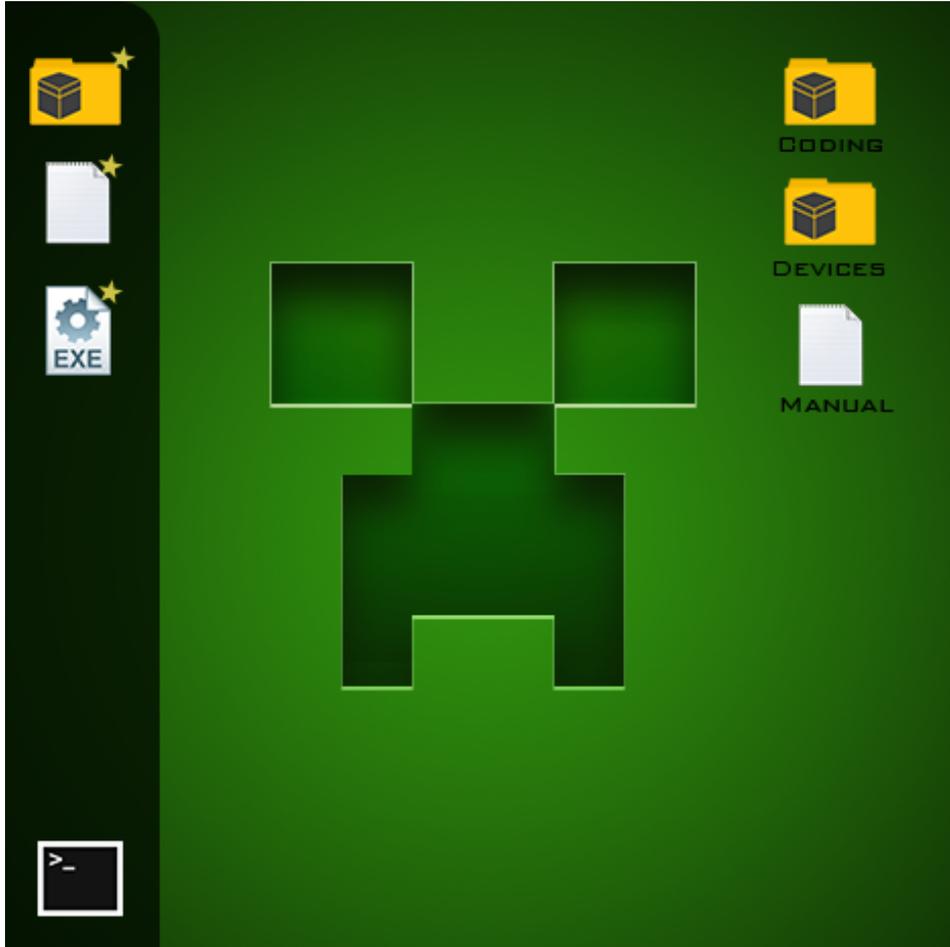
#### Crafting:

Tin Plate	Tin Plate	Black Dye
	Piston	Repeater
Tin plate	Tin Plate	Advanced Electronic Circuit

The Printer, opposite of the Shredder, is used to make items from the network. It converts digits and uses ink to make the item that was first disposed of.

### 3.18 Computer Tier 2

The Computer of the Second Age is a compact and upgraded version of the First Age Computer. Apart from space, the first change you will notice is the GUI in the monitor:



As can be seen in the figure above there are 4 icons. 3 of them have a star. The icons with a star mean "Create New". You can move the icons around and place them where you want.

- Command Prompt



This Icon opens the Command Prompt like the standard First Age Screen. Use the command "Start CTDOS to come back to this GUI.

- Text File



A text file is not all that special. It's only use is to write something like a note. Useful for TODO lists ore explanation to someone else who uses your programs.

- Folder

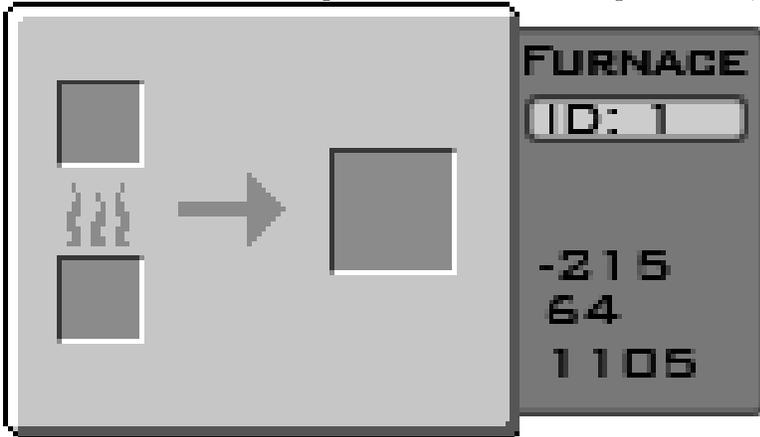


A folder can be used to store text files and program files. Folders on the desktop can be found under:

"CTDOS/Desktop/<Folder Name>"

There is one folders that will always be available on the CTD OS computer; "Devices".

In this folder you can find all Devices connected to the network. You can also find possible inserted Disc Drives here. Clicking on such devices can open a GUI, for instance a furnace:



Also from here you can change the ID of the block. You can also see what items are inside the block, but you cannot interact with those items. So you can't manually place or pull items at your monitor.

- Program File



Program files are the most important ones in the CTD System. In these files you can write the programs that will be used. It is possible to use multiple files for a single program.

In case you want to you need to refer to the location of that file.

"./" Will set you in the current directory, or you can use "CTDOS/Desktop/.." To find it.

e.g. if your program file is in "CTDOS/Desktop/Coding/MyFirstProgramFile.exe" and you have second program file there you can refer to it as "CTDOS/Desktop/Coding/SecondFile.exe" or "./SecondFile.exe"

Using more dots will make you go up one file. "../" would bring you outside the Coding Folder and onto the Desktop Folder.

Assuming the second file has a function called sayHello(), you can use it by doing the following:

```
import "./SecondFile.exe";  
  
SecondFile.sayHello();
```

The commands you can use can be found in chapter 6.3. [Click Here](#)

## 4 Third Age

### 4.1 Distronium Cube

#### Crafting:

Distronium Crystal	Distronium Crystal	Distronium Crystal
Distronium Crystal	Distronium Crystal	Distronium Crystal
Obsidian	Diamond	Obsidian

A Block with GUI that distorts the Ender Power Source, allowing you to change form of ender pearls by fusing it with another item.

Ender Pearls	Copper Wire	Enderium Wire
	Iron Ingot	Enderium Ingot
	Diamond	Enderium Crystal

### 4.2 Enderium Circuit

#### Crafting (Soldering Station):

Advanced Electronic Circuit	Copper Wire	Advanced Electronic Circuit
Copper Wire	Enderium Crystal	Enderium Wire
Advanced Electronic Circuit	Enderium Wire	Advanced Electronic Circuit

Crafting Component. Can also be used on certain blocks to make them connect-able to the network.

### 4.3 Enderium Cable

#### Crafting:

Enderium Wire	Enderium Circuit	Enderium Wire
Redstone Dust	Redstone Dust	Redstone Dust
Enderium Wire	Enderium Circuit	Enderium Wire

**Makes:** 8 Enderium Cables Can be used to instantly transfer items, power and commands. Can only be operated in a Third Age Network.

#### 4.4 Ender Receiver

**Crafting:**

Ender Pearl		
Enderium Circuit		

Crafting Component.

#### 4.5 Ender Transmitter

**Crafting:**

Ender Receiver	Redstone Torch	

Crafting Component.

#### 4.6 Wireless Modem

**Crafting:**

Ender Receiver		Ender Transmitter
Tin Plate	Enderium Circuit	Tin Plate
Tin Plate	Tin Plate	Tin Plate

Used to wirelessly connect Gates to each other.

## 4.7 Gate

### Crafting:

Sticky Piston	Enderium Circuit	Ender Receiver
Enderium Cable	Distonium Crystal	Enderium Crystal
Sticky Piston	Enderium Circuit	Ender Transmitter

The Gate has the latest priority in a non-set destination pipe system. So if an item is seeking for an available inventory slot, it will enter the Gate only if there is nothing available. Gates are used to transport items, energy and commands over long distances without any cabling between them.

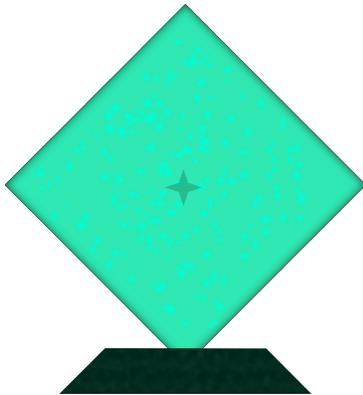
## 4.8 The Cube

### Crafting:

Block of Distonium	Block of Distonium	Block of Distonium
Block of Distonium	Nether Star Block	Block of Distonium
Enderium Block	Enderium Circuit	Enderium Block

The Cube has a variable size. It's 3x3x3 at smallest and 15x15x15 at largest.

This is not because you must built a multiblock, but the block can grow. The Cube can contain up to 2.097.152 items. That means for every 175k items, the cube will have grown 1x1x1 block.



Items can be inserted already simply by tossing, but if you want to connect to it, you must connect to the bottom center part of the underlay.

For coding, the Cube works the same as any regular chest, except it can hold more and needs power to be accessible.

## 4.9 UDB port

### Crafting:

		Tin Plate
Enderium Circuit	Sticky Piston	
		Tin Plate

Crafting component.

## 4.10 UDB

### Crafting:

	Comparator	Memory Chip
Tin Plate	Sticky Piston	Enderium Circuit
	Redstone Repeater	Memory Chip

Upgraded version of the SqDisc. *<Can hold up to 4x more bytes.>*

## 4.11 Laptop

### Crafting:

Tin Plate	Monitor Tier 2	Tin Plate
Wireless Modem	Enderium Circuit	UDB Port
Gate	Mother Board	Memory Chip

A Laptop is the same as a tier 2 monitor, but then used wireless. When you start it up, it will search for nearby Wireless Modems and then it's up to you which you connect to. use "connect <Insert Modem ID>" to connect to a modem and thus a network.

## 5 Tools and Utils

### 5.1 Calculator

#### Crafting:

Iron Ingot	Glass	Button
Iron Ingot	ALU	Iron Ingot
Iron Ingot	Iron Ingot	Iron Ingot

Right click a block. The Calculator will register it's X, Y and Z Coordinates until you click the next block. When you do, the Calculator will print the following message:

Distance X:  $x_2 - x_1$

Distance Y:  $y_2 - y_1$

Distance Z:  $z_2 - z_1$

Total Blocks:  $X * Y * Z$

### 5.2 Hammer

#### Crafting:

Iron Ingot	Iron Ingot	Iron Ingot
Iron Stick	Stick	Iron Ingot
	Stick	

Place with an Ingot in crafting slot to make a plate. Can be used 192 times.

### 5.3 Security Station

#### Crafting:

Iron plate	Disk Driver	Iron plate
Advanced Electrical Circuit	Zombie Head	Golden Sword
Iron plate	Memory Chip	Iron plate

The Security Station is a special block that can secure your network. With your security system active you can set password to any access point in your network (every monitor and Wireless Router). A player needs to interact with the network at least once before it's name pops up in the Security Station, regardless of a failed attempt or success. Then in your security station you can "Allow" or "Block" Players. Allowed Players can access your Network without password. Blocked Players will receive 1 heart Damage for each attempt, and they will not regenerate that health as long as they are near the device.

Any Player without password can still access your network through **Hacking**. For anyone who is not white-listed, they have an extra button in the login screen "Hack". When hacking the player enters a mini game. When they win, all their access logs on the Security Station will be deleted and are put inside your white-list.

*<The Mini Game and Security Upgrades will be explained upon updates!>*

Whitelisted people can access your network, but cannot edit or delete any programs. They can however make new programs or upload new programs through SqDiscs or UDB. They can also see the coordinates and other information in your "Devices" Folder.

*<Whitelist Restriction and security also explained upon updates!>*

## 6 Code

### 6.1 Basic Coding

*<if, while, parameters, variables, ect.>*

#### 6.1.1 If statement

```
boolean a = true;
boolean b = false;

if(a)//true
    print("a is true");
if(!b)//b = false , -> !b = if b == false
    print("b is false");

if(a && !b) { // multi line if
    print("a is true and...");
    print("b is false");
}
```

```
a is true
b is false
a is true and...
b is false
```

#### 6.1.2 While statement

#### 6.1.3 For statement

#### 6.1.4 Parameters and Variables

### 6.2 Advanced Coding

*<arrays, conversions, function(), ect.>*

### 6.3 Commands

```
ls
```

ls is short for listing. It returns the names and ID's of all machines that are hooked up to the network like this:

```
1 Furnace          2 Chest
3 Chest           4 Ender Chest
5 Furnace         6 Chest
...
294 Enchantment Table 295 Furnace
```

You can get the same screen with the second and third age computer by using the ls command in the Command Prompt program.

```
getBlock(1);  
getBlock("Vanilla.Furnace");  
getBlock("Furnace");
```

1	Furnace	<-215, 64, 1105>
1	Furnace	<-215, 64, 1105>
5	Furnace	<-214, 64, 1105>
94	Furnace	<-209, 64, 1120>
1	Furnace	<-215, 64, 1105>
5	Furnace	<-214, 64, 1105>
9	Redstone Furnace	<-215, 64, 1105>
23	Iron Furnace	<-215, 64, 1105>
94	Furnace	<-209, 64, 1120>
101	Electrical Furnace	<-215, 64, 1105>
153	Alloy Furnace	<-215, 64, 1105>

If there is no further information given, the `getBlock()` Command will return the results as seen in the table above. You can filter the output by ID, Block name and even filter out specific Mods. The only information this command holds are the coordinates of the selected block. This information however, can be used by other commands.

```
getBlock(451).getRedstone();
getBlock(451).getRedstone().getStrength();
getBlock("Vanilla.Furnace").getRedstone();
```

Suppose 451 is a Redstone Receiver that receives a redstone power of 10, the program would return:

451	Redstone Receiver	<-209, 62, 1097>	true
451	Redstone Receiver	<-209, 62, 1097>	10
1	Furnace	<-215, 64, 1105>	false
5	Furnace	<-214, 64, 1105>	false
94	Furnace	<-209, 64, 1120>	false

The furnaces are false, since they can't really receive a redstone signal. This cannot be checked with the First Age Computer, since it cannot receive, nor send any redstone signals through the electrical circuit.

```
getBlock(452).setRedstone(true);
getBlock(452).setRedstone(true).setStrength(10);
getBlock("Vanilla.Furnace").setRedstone(true);
getBlock(452).setRedstone(false);
```

The setRedstone commands will not output text to the monitor.

Suppose 452 is a Redstone Emitter, then with the first line, the Emitter would be set to true and send a redstone power of 15 to adjacent blocks.

The second line would set the Emitter to send a redstone power of 10 to adjacent blocks.

Since the Furnace cannot send any redstone power, the 3th line will end up being an **<ERROR>** and be ignored.

The last line, back to the Redstone Emitter will set it's redstone power to 0, and thus off.

```
print("Hello Minecraft!");

String text = "Bye Minecraft!";
print(text);

print("CookieZ").Color(Red);

print("R".Color(Red) + "A".Color(Blue) + "I".Color(Cyan) + "N".Color(Yellow) + "B".Color(Orange) +
      "O".Color(Green) + "W".Color(Violet));
```

```
Hello Minecraft!
Bye Minecraft!
CookieZ
RAINBOW
```

The Print Command, prints text to the monitor. In the First Age colors are not available.

## !NOTE!

Commands from this point will not be available for the First Age Computer.

```
label getBlock(4) EChest
```

using `getBlock(4)` can be kind of confussing, especially if you have over 200 ID's to remember and you don't want to scroll down the ls list everytime. By using `label`, you can edit the ID of the block into a custom ID. In the code above we change the ID of the block with ID 4, into `EChest`. When we then do `getBlock(EChest)`, we can expect the following result:

```
EChest  Ender Chest  <-213, 64, 1105>
```

These kind of features don't have any functionality on their own, but does improve the readability of you code.

```
getBlock(1).getSlot(2);
getBlock(1).getSlot(2).push(getItem("Coal"));
getBlock(1).getSlot(2).push(getItem("@Coal"));
getBlock(1).getSlot(3).pull();
getBlock(1).getSlot(2).push(getBlock(3).getItem("@Coal"));
getBlock(1).getSlot(3).pull().setDestination(getBlock(3));
```

Assume we are now interacting with a furnace. The Furnace has 3 slots. 1 is the "Burnable" slot. This is the slot for items that will be melted down in the furnace. Slot 2 is the "Fuel" slot where items such as (char)coal and wood goes. Then Slot 3 is the "Results" Slot where the results of the burning process appears. The Push and Pull Commands will not show any text, but the command `getSlot` will return the item and amount of the selected slot.

```
Coal  21
```

The `pushSlot` command will try and push items into that slot. In line 2, we try to push into slot 2, but only an item with the name "Coal" in it. Again as with blocks you can filter on mod, but you can also make it litteral. Even if there are no other mods installed, the computer will try to push both Coal and Charcoal if it can find one. In line 3 however we added an @ sign to filter on the precise name Coal.

These two commands however will pick a "Coal" from a the first inventory it encounters, since a specific inventory has not been defined.

Line 4 is a pull Command from the results slot. This will pull any item from this slot and place it in the first available inventory space, since this has not been specified.

Line 5 and 6 have a better specification. Line 5 pushes only Coal from the specified inventory 3. This inventory happens to be a chest. This command will not try to pull coal from any other inventory even when there is no coal inside the specified chest.

Line 6 will pull any item from slot 3 and push them to it's destination `getBlock(3)` to the first available slot. Of Course you can again be more specific by `getBlock(3).pushSlot(x)` and fill for x the slot you want to push it into.

If the item destination is full this will result in an **<ERROR>** and will be ignored.

**Hint:** Though the program will be ignored, if you want to play safe you can use tricks, such as:

```

Slot brn =getBlock(1).getSlot(1);
Slot fuel =getBlock(1).getSlot(2);
Destination des =getBlock(3);
while(true){ //Makes the program always run
    if(brn < 64) { //Slot 1full?
        brn.push(getBlock(6).getItem(@Vanilla."Oak Wood"));
    }
    if(fuel < 64) { //Slot 2full?
        des = fuel; //Set destination on slot 2
    } else { //if Slot 2full
        des = getBlock(3); //set destination block(3) =chest
    }

    result.pull().setDestination(des);
    //Pull the result from slot 3 and send it to it's set location
} //Repeat

```

The Code above is an example on how to make a charcoal farm using these computers. If there can be burned something, it will pull oak wood from the chest with ID 6 and put it in Slot 2, ready to be burned.

Then it will check if slot 1 is full on fuel. If not, the destination of the result, in this case charcoal will be set to slot 2. If slot 1 is full the destination is set, or reset to the chest with ID 3.

In the end it will pull the item from slot 3 and sent it to either Slot 2 of the furnace, or an available slot in the specified chest, depending on the condition we just mentioned.

This is a basic example. It's not fuel efficient nor wood efficient. If you want to be as efficient as possible you check the amount of charcoal you have (you don't need to have a chest full of charcoal, right?) and you check on the amount of wood in slot 1 before you push a charcoal, since 1 charcoal can burn 8 items.

```

String[] ores ={"ironOre", "goldOre", "Coal"};
String[] food ={"apple", "bread"};
String[] farm ={"seeds", "saplings"};

Block filter =getBlock(filter); //ID of filter pipe set to filter
filter.setNorth(ores).oreDirectory(true);
filter.setEast(food);
filter.setSouth(farm).metaData(false);

```

Here the sides of a filter pipe are set. North is set to allow iron ore, gold ore en coal. Nether ores and such are also included since the oreDirectory is set to true.

The East side is set to apples and bread, and the South side redirects the farm items, Seeds and Saplings.

However there are "oakSapling", "birchSapling", "spruceSapling", ect. the metaData has been set to false, so all these saplings will be redirected.

## !NOTE!

Commands from this point will not be available for the Second and First Age Computer.

```
Block capacitor =getBlock(372); //Assuming block 372 can use or store power
capacitor.setRFT(1600);
capacitor.getRF();
```

Using the second line you can set the RF/t rate that is allowed on the Block. Be aware though. If you set it higher than the block can handle, the system will still try to force that amount of RF inside the machine in a tick, causing the machine to explode.

The Second line will return the following: This means, the capacitor has at the moment of the

```
372 Capacitor <-198, 60, 1109> 3.504.382 1600 RF/t
```

function call over 3.5 Million RF, and it's RF/t has been capped at 1.6k RF/t. However the RF/t tab only shows the maximum RF/t rate. If the actual RF/t is lower at that moment, due to a lack in power, this will not be shown.

```
Block wifi =get Block(458); //Assuming block 458 is a Wireless Modem
wifi.setFrequency(21105);
wifi.getConnection();
```

Here you see an example on how to set the Frequency of a Wireless Modem. This Modem's frequency is set to 21105, but it is possible more Modem's have that frequency. Modem's with the same Frequency will be connected. The Third line will return the Modem with all it's connections:

```
458 Wireless Modem (21105) 469
                           473
                           474
                           481
```