I'm not a bot

Skip to main content Reddit and its partners use cookies and similar technologies to provide you with a better experience. By accepting all cookies, you agree to our use of cookies to deliver and maintain our services and site, improve the quality of Reddit, personalize Reddit content and advertising, and measure the effectiveness of advertising. By rejecting non-essential cookies, Reddit may still use certain cookies to ensure the proper functionality of our platform. For more information, please see our Cookie Notice and our Privacy Policy. Legacy Green Slime Crystal Slime Crystals are one of many materials used to craft and maintain tools in Tinker's Construct. It is a non-metallic material that can be worked directly from the Part Builder. Using Slime Crystals to make tool parts gives tools the Slimy ability, giving it greatly increased durability And the ability to randomly spawn small blue slimes when fighting. Crafting[] A Slime Crystal is made by smelting Slimy Mud in a Furnace. The resulting crystal will be either blue or green, depending on if the player used Gelatinous Slime or vanilla Slimeballs. Green Slime Crystal Usage[] Slime Crystals can be used in the Part Builder, making any part from a pattern. Slimy tool parts have high durability, low reinforcement, high handle modifier, and good speed. However, this comes at the cost of not being able to mine higher-tier materials. However, because of blue slime's excellent handle modifier (Best for any non-metallic material, and the second best handle modifier overall, at par with Ardite and just below Manyullyn) it is commonly used for tool handles. Both blue and green slime tool parts can be found in village tinkerers. Luma - A lighting Mod for Forge Luma is a mod for Minecraft Forge all about Lighting. It's in a very early stage right now, but expect it to grow a lot over the next months. Current version: 0.2.2 (MC 1.6.2 Forge 9.10.0.804) direct download Changes in this version: [NEW] Added colored Fixtures [FIX] Moved Fixtures down a bit when displayed as Inventory Items Optimised render code Changed standard starting IDs (Block: 3750 / Item: 12050) - can be changed in the config file Changelog: 0.2.2 - 2013-09-04 - [NEW] Added colored Fixtures - [FIX] Moved Fixtures down a bit when displayed as Inventory Items - Optimised render code - Changed standard starting IDs (Block: 3750 / Item: 12050) - can be changed in the config file 0.2.1 - 2013-09-01 - [NEW] Added custom tab "Luma" to creative mode inventory. All Luma blocks/items moved there - [NEW] New custom Renderer for Luma Lamps / Multilamps. disables Shading and makes the lamps glow more - [NEW] Added Debug Item only accessible in creative mode/NEI. Shows information in chat about the block that was right clicked. Will probably be removed at v1.0 - [FIX] "Off"-states of blocks were shown in NEI. Fixed by disabling them using the NEI API - [FIX] Tweaked bar textures for caged lamps 0.2.0 - 2013-08-27 - [NEW] New Block: Luma Fixtures (only white for now) - [FIX] Corrected Textures for Multilamp - No longer has strange artifacts in corners - More internal code cleanup 0.1.1 - 2013-08-19 - [FIX] Levers can now be placed directly on lamps - Transferred code to GIT repository - Internal code cleanup / refactoring 0.1.0 - 2013-08-18 - First Release Current features: Luma Lamps: Dyeable Lights, designed to provide a clean, almost industrial looking light source. Luma Lamps output the maximum possible light level and are powered by redstone. They come in normal and inverted varieties, as well as plain and caged versions. Screenshot: Recipes: Luma Lamp Caged Luma Lamp Inverted Luma Lamp Inverted Caged Luma Lamp Luma Fixture Luma Fixtures are wall (and floor/ceiling) mountable lamps, powered by redstone. They come in caged and inverted versions. Their placement behaviour corresponds to torches. That means, if the block they're placed on is destroyed, they will drop as items. Comes in all 16 colors (color white with dyes, like wool) Screenshot: Recipes: Luma Fixture (4x) Inverted Luma Fixture (4x) Caged Luma Fixture (4x) Caged Inverted Luma Fixture (4x) Multilamp: Changes its color according to the strength of Redstone power it receives. This is especially useful when used with other mods like Computercraft which can output different power levels. It comes in a normal and a caged version. The Multilamp requires a new item called Chroma Stone in its crafting recipe. Screenshot: Recipes: Chroma Stone Multilamp Caged Multilamp Please let me know what you think. This is my first mod. I'm a programmer but a noob in Java/Forge API so please be gentle, I'm still learning Direct Download of current version (0.2.2): direct download More Information: www.kelmai.com/luma Introduction Video: Like my mod? Help me pay for hosting: Reactions: Geometry, frederikam, AlanEsh and 4 others Looks useful. Combining this with MFR rednet cables, I can see some advanced control panel designs. Or a player-activated seizure bomb. Either build is equally useful. The only complaint I have is that the blocks don't join well when they're different colors, creating a visual artifact. Reactions: the_j485 Yes, MFR, of course! That would work great! You could do some interesting stuff with the wave generator... I'm sure you're talking about the corners of the multilamp. I noticed that too, it's because of the bevel effect on the texture. I'll adjust that for the next release, so the dieamond shapes in the corners will look as smooth as the ones in the middle. Reactions: RedBoss and PeggleFrank Update 0.2.0 released Direct Download Changes: [NEW] New Block: Luma Fixtures (only white for now) [FIX] Corrected Textures for Multilamp - No longer has strange artifacts in corners More internal code cleanup What does a Multilamp look like with no signal? Black? Do you have compatibility with the Forge Multipart API? I could see a particular use of a caged blue light on a 2 x 2 box 3 high and containing a Dimension Door... but only if it can get centered in the middle of the intersection of the four blocks... Do you have compatibility with the Forge Multipart API? I could see a particular use of a caged blue light on a 2 x 2 box 3 high and containing a Dimension Door... but only if it can get centered in the middle of the intersection of the four blocks... So, it is bigger on the inside then? Sounds like a blue box. Reactions: RedBoss Do you have compatibility with the Forge Multipart API? I could see a particular use of a caged blue light on a 2 x 2 box 3 high and containing a Dimension Door... but only if it can get centered in the middle of the intersection of the four blocks... I'm actually looking at this right now. Expect Multipart integration in one of the next releases! Reactions: PeggleFrank I don't think that microblocks are supposed to be functional. Imagine the problems, you've got a lamp cover on one face, a computer cover on another, a computer monitor, a piston, and a glowstone cover on the other faces, what is that block supposed to do? What happens when you put a redstone block panel on the last face? Does the redstone panel give the computer and the lamp a redstone signal? And what does the piston panel do? To be honest, I was surprised when I found that glowstone panels emitted light. I should try making a glowstone lamp panel, and see if that responds to a redstone signal. Do you think that you will be able to make your lamps into panels and strips and notches that turn on and off? p.s. your mod looks really nice! I don't think that microblocks are supposed to be functional. Imagine the problems, you've got a lamp cover on one face, a computer cover on another, a computer monitor, a piston, and a glowstone cover on the other faces, what is that block supposed to do? What happens when you put a redstone block panel on the last face? Does the redstone panel give the computer and the lamp a redstone signal? And what does the piston panel do? To be honest, I was surprised when I found that glowstone panels emitted light. I should try making a glowstone lamp panel, and see if that responds to a redstone signal. Do you think that you will be able to make your lamps into panels and strips and notches that turn on and off? p.s. your mod looks really nice! Thanks I thought that the microblocks would just be decorative, without functionality/light emission. But since you mentioned that glowstone microblocks emit light, I'll have to look into that. Can't promise anything... This looks like a great mod, I really like these smooth textures on the regular luma lamps. I'd even go as far as to suggest you could replace the multilamp texture with a less-textured/smoother texture. When I saw this thread, the first thing which sprang to mind was an old build I made with RP lamps. It's a very narrow use case, but it did work extremely well with RP lamps. A bit less so with your mod. As you can see in picture 1, the RP lamps light effect is a bit bigger than the lamp itself, making the stepped on pressure plates go inside the lights' boundaries. The second picture where I tested the Luma lamps doesn't have this same effect. Picture 1: Disco Floor with Redpower - If you run fast a lot of pressure plates are pressed! Picture 2: Disco Floor with Luma Now I don't know if it is possible to add a similar effect (it doesn't have to be the same solution as RP), but if you could the results would be awesome! A side-note, when testing the mod in creative mode I noticed there is no separate Luma lightning tab yet in the creative inventory. This could be a handy addition. These are the things I noticed in a quick test, there's no pressure on implementing them, I'm just giving some input. Keep up the good work! Thanks for the great feedback! It's always good to get constructive input! I was looking for some sort of smooth lamp without too much texturing on it. So why not just make a mod myself? Yes, I know the RP lamps. They were (obviously) an inspiration for this mod. I have been thinking about creating this "glow" as well. I just looked at the Tesselator and it's quite easily possible with a second render pass. I wanted to do it somehow differently because i didn't just want to copy RP. But I think there is no better way to make the lights glow. So I think this will be added in one of the next updates. I'll probably put an option into the config file to turn it off, because i'm sure not everyone likes this kind of thing. I might put a cleaner texture on the multilamp at some point. I agree, it looks a little "busy". I'm glad you like it! Please post about any bugs you might find or ideas you might have! I already said this in another thread about a similar mod "idea" (but your mod is done, and it seems awesome, the whole changing colors lamp is amazing), but I think that it woud be coo to have a special kind of lamp that iluminated a big area AND had somekind of special ability in that area (eg, put mobs on fire). I would also like to see a lamp that instead of lighting things, it makes things darker. Although that may be very hard to find. Anyway, keep on the good work! I already said this in another thread about a similar mod "idea" (but your mod is done, and it seems awesome, the whole changing colors lamp is amazing), but I think that it woud be coo to have a special kind of lamp that iluminated a big area AND had somekind of special ability in that area (eg, put mobs on fire). I would also like to see a lamp that instead of lighting things, it makes things darker. Although that may be very hard to find. Anyway, keep on the good work! I've been thinking about large area lighting. The Wrath Lamp from Factorization does that, but is quite "hacky" and buggy and can produce a lot of lag. Sadly I think the way this is done is the only way of achieving this. I think this won't be in my mod, unless I find a better way of doing it. Adding special effects in the area is an interesting idea and wouldn't be a problem (i think...) The "Shadow Lamp" is a very interesting idea, I'll have to look into this, it might be possible! Update 0.2.1 released Direct Download Changes: - [NEW] Added custom tab "Luma" to creative mode inventory. All Luma blocks/items moved there - [NEW] New custom Renderer for Luma Lamps / Multilamps. disables Shading and makes the lamps glow more - [NEW] Added Debug Item only accessible in creative mode/NEI. Shows information in chat about the block that was right clicked. Will probably be removed at v1.0 - [FIX] "Off"-states of blocks were shown in NEI. Fixed by disabling them using the NEI API - [FIX] Tweaked bar textures for caged lamps This is unbelievably cool. Especially the multilamp. It'd be cool if you could make some multiblock street lamp or something of the sort. Reactions: dgdas9 Page 2 The multilamp looks neat, I can do some projects with it in CC but it's too expensive considering, but it's too expensive considering the that it's an aesthetic functionality. There is a very good probability that when I update ShneekeyCraft to a new version, that this mod will be included. It fills a niche which is sorely missing so far. Yes, go ahead, include it! I'm glad you like it! This is unbelievably cool. Especially the multilamp. It'd be cool if you could make some multiblock street lamp or something of the sort. Thanks a lot! I'll have to finish a few besics first, before adding more complex stuff. But thanks for the input! The multilamp looks neat, I can do some projects with it in CC but it's too expensive considering, but it's too expensive considering the that it's an aesthetic functionality. Do you think so? I was wondering if it still was too cheap, at first I had a recipe with 4 Chroma stones (4 diamonds)... You're right, it's mostly aesthetic but has many practical uses as well (multi colored indicators for example). I think its functionality is quite advanced, so it should be worth the diamond (also, it makes sense construction wise. You need to break the light into individual colors somehow...). What do you guys think? should the multilamp be cheaper? Or more expensive? Or is it just right? Do you think so? I was wondering if it still was too cheap, at first I had a recipe with 4 Chroma stones (4 diamonds)... You're right, it's mostly aesthetic but has many practical uses as well (multi colored indicators for example). I think its functionality is quite advanced, so it should be worth the diamond (also, it makes sense construction wise. You need to break the light into individual colors somehow...). What do you guys think? should the multilamp be cheaper? Or more expensive? Or is it just right? Although I may use it late-game anyway, I'd gladly make use of these if you'd use one diamond for four lamps. Keep up the good work! Update 0.2.2 released Direct Download Changes: - [NEW] Added colored Fixtures - [FIX] Moved Fixtures down a bit when displayed as Inventory Items - Optimised render code - Changed standard starting IDs (Block: 3750 / Item: 12050) - can be changed in the config file The caged lamps are a bit odd, they're basically just a lamp with stripes on them. Are there any plans to make them look a bit more like the RedPower 2 Cage Lamp? *Edit* D'oh, just spotted the "Cage Fixture". Nice! Yes, the caged fixtures should do the job although they are also only fixtures with "stripes" on them. I personally like it when Minecraft blocks look very simple and blocky. I don't like when blocks look too complex (The RP Fixtures are ok, though. I just like to keep it simpler for mine). That's why I keep my blocks as simple as possible. For example, I don't like the Bibliocraft tables and lamps at all because they don't look "blocky". That doesn't fit in with the rest of Minecraft. I also don't like high res textures mixed with vanilla minecraft (unless you use a high res texture pack for everything, then it fits with the rest). I know some people like it differently, but you can never make everyone happy Reactions: PeggleFrank I have been thinking about creating this "glow" as well. I just looked at the Tesselator and it's quite easily possible with a second render pass. I wanted to do it somehow differently because i didn't just want to copy RP. But I think there is no better way to make the lights glow. So I think this will be added in one of the next updates. I'll probably put an option into the config file to turn it off, because i'm sure not everyone likes this kind of thing. Instead of a config option, just add one more recipe that turns any colored light into a glowy light -- like apply a slime ball to any light. This way we can have or not have the glow on a per light basis Instead of a config option, just add one more recipe that turns any colored light into a glowy light -- like apply a slime ball to any light. This way we can have or not have the glow on a per light basis That's actually a great idea, I'll probably do something like that. I'll also include config options to individually enable or disable each lamp type (standard: all enabled). This way you can pick and choose what you like. This should come in the next update. Ok got a question for you Is it possible to send a single redstone signal to a multi lamp and that multi lamp pass the redstone signal through itself to an attached lamp decreasing the redstone power by 1. So in effect you can power a single row of lamps from 1 redstone signal and it change color as it runs down the row of lamps? I don't know if it's possible to do but it would open up so many opportunities where redstone signals aren't needed against every block. Ok got a question for you Is it possible to send a single redstone signal to a multi lamp and that multi lamp pass the redstone signal through itself to an attached lamp decreasing the redstone power by 1. So in effect you can power a single row of lamps from 1 redstone signal and it change color as it runs down the row of lamps? Could you get that effect by running redstone dust along the top of the lamps? That would only work for horizontal lines, of course, but having signal leak by default would make these lamps un-usable for a large number of use cases e.g. display boards, tetris games, pictures, etc. hmm perhaps a new set of "Redstone wired lamps"? What will a multilamp do if it gets two different redstone signal strengths from different directions? Having that Multilamp means I can finally make a giant color monitor! Time to make some computercraft programs for 8-bit games. Would you mind doing a quick update to 1.6.4? I would like to see either forge multipart integration or some kind of glowy strips that you can stick on the wall to light it up. (it would have a lower light level than a full block lamp). Otherwise keep up the good work and have fun modding Would you mind doing a quick update to 1.6.4? Hi guys! I'm sorry for the lack of updates recently, real life intervened a little bit I hope to update it by the end of december. If I don't find the time by then I'll probably consider giving it to someone else to continue but I'm pretty sure I'll be able to do it. Reactions: AlanEsh Page 3 The FTB Forum is now read-only, and is here as an archive. To participate in our community discussions, please join our Discord! You are using an out of date browser. It may not display this or other websites correctly.You should upgrade or use an alternative browser. Hello, Is there a way to automate Hardened Glass with Tinker Smeltery ? Hardened Glass is crafted with an obsidian in a casting bassin and 1/2 ingot of lead in a smeltery. I have no problem on how to put or get liquid/items in the smeltery/casting bassin but my problem is that when I ask my AE2 system to craft a Hardened Glass,it exports Obsidian and Lead in a chest that is after inserted into smeltery with EnderIO conducts BUT lead is inserted in my casting bassing even if my Obsidian is not in my casting bassin yet. What I want is that liquids in the smeltery are exported only if there is an item in my casting bassin. Comparator from Vanilla or RedPower don't detect there is an item in the casting bassin. A solution may be SFM but I don't want to see all that inventory cables and I want it discreet. Typically with Hardened Glass my pattern is 1 Lead with 2 Obsidian gives 2 Hardened Glass. Lead is smelted in the smeltery. In the same time Obsidian is inserted in my casting bassin. Then liquid lead is poured in the casting bassin. Hardened Glass is imported in my AE2 system. Then liquid lead is poured in my casting bassin before the second Obsidian is inserted. Then my system is stuck. It is worse when my obsidian needs to be crafted or is missing. Lead go to the casting bassin before the first obsidian is. Thanks a lot if you have an idea. Pretty easily Have an interface going into a chest with the crafting pattern using 1 lead + 2 obsidian = 2 hardened glass Then have 2 filtered conduits (of your choice) pulling out of the chest, 1 that pulls lead, the other pulls obsidian. The obsidian conduit leading to the basin, the lead leading to the smeltery. Then we are on to automating the smeltery... Now this I am less sure on, but if I am correct SHOULD work. Steves factory manager, redstone emitter next to the faucet, and inventory cable next to the basin. Have SFM check if the basin contains obsidian, if true, pour activate the redstone emitter. If false, no redstone signal. When a faucet recieves a redstone pulse it will pour... and if the signal stays on it will keep pouring until it is done. And won't pour again until the redstone signal is activated again. Reactions: pizzawolf14 My usual answer to this would revolve around the usual "stick a comparator on the thingy and use the redstone signal to tell the other thingy to do the thing" method, but I don't think casting basins give a signal like a normal inventory. I could be wrong though... Thanks for the answer jordsta95. But as I said, I want to avoid SFM because there is no way to hide this horrible inventory cables that need to be close to the casting bassin/smeltery controller/smeltery drain and I hate to see cables. AE2 cables can be hide by facades with the assembly tables, enderio conduits by facades with the painting machine but the inventory cables it is very difficult and ugly. Typically, I can't have an enderio conduit under the drain to pour the liquid and an inventory cable. So I have to place it next that is not nice. And to answer to malicious_bloke, tested in creative mode, casting bassin don't give a signal even with comparator in RedPower2. All other solutions are welcome. PS : I don't hate SFM, I use it for my AE2 inscribers and charger but I don't like how hard it is to hide the cables. Is there a specific reason you want to do that using a smeltery? Automation of Hardened Glass production using an Induction Smelter is pretty easy. AE2 level emitters might help here if you are willing to make a rube goldberg machine. Principally you need to have an AE2 subnetwork that is attached to an interface that tries to export obsidian into the hardened glass network. A level emitter configured to detect obsidian will emit a redstone signal that you will use to disable the importing of obsidian. So you should have exactly ONE piece of obsidian in your AE sub network. An export bus will then be configured to export obsidian into the casting basin. The level emitter will go low, so a timer will be needed to continue inhibiting obsidian for a while. The level emitter going low also detects that the obsidian that WAS in the network MUST be in the casting basin, so another latch will then turn on the flow of lead. An import bus will extract the glass when finished, and another level emitter can detect the presence of that in the AE network and reset all the latches to their ready to craft state (with the flow of lead turned off). There might be a quantity of lead accidentally poured into the basin, so you'll need to have a duct set up to extract that. hmmm. Rube will be proud. Nice Chris Becke but it is huge and cause frequent update. Furthermore, I don't want to craft only hardened glass but other items with casting bassin too (tesseract, energy cells...). So one level emitter for each item it is huge. Thanks a lot Eldiwoodio I will see for that camouflage. And I don't use induction smelter because it needs obsidian dusts then an addiitional operation. Furthermore, the problem is the same with tesseracts which have to be created this way. Bad news , I just try in creative mode with SFM and when obsidian is in the casting bassin the condition in the manager is false. When I switch with a chest it is true... I will try with Create Liquid Input/output so operations are sequential, maybe it will works. Edit : Second test : storage bus don't work with casting bassin Precision : My ModPack is FTB Infinity Last edited: May 29, 2015 I would try computercraft. Have it push obsidian from the chest, wait a moment, then trigger the spout. I'd also tap it into the controller so it could run only when the smeltery has fluid lead in it. If you want the easy solution- just use a vanilla hopper to feed the casting basin. I didn't think about Computercraft. I will try this And hoppers don't guarantee that lead is not poured after obsidian Try using a buildcraft cobble structure pipe with a "Gate" on it. When placed next to any object it will automatically detect if it has an inventory. You can then specify if you want it to send a redstone signal when the container has an item, when it is half full, or when it is full. With this you can automate the placing of obsidian and pulsing of the faucet. 1: Place pipe and gate adjacent to basin 2: Set settings on gate to give off a redstone signal on inventory full 3: Use Rednet cables or redstone conduits to send a signal from the gate to the faucet (if it doesn't work set signal to "Strong") You will have to automate lead into the smelter but it will stop pouring fluids once there is no obsidian. Forgot about BC Gates. I don't know that they can detect the inventory status of the basin. However, you could use an autarchic and a basic gate with a timer, to run the system. Have a wooden pipe between a chest and the basin, with an autarchic gate on it. Set it pulse once if there's obsidian in the inventory of the chest, and send a red pipe signal up to the gate on the structure pipe. That gate takes the red pipe signal and starts the timer. Work out how long the Obsidian takes to move from the chest to the basin, and then set the timer for that long. At the end of the timer, it gives a redstone signal to pour in the lead. If there's going to be something besides obsidian in the chest, use an emerald pipe instead of a wooden one so you can filter the output. If you do use BC gates, Gold and higher Gates can specify specific items to look for in an inventory. You can build a 1x1 internal smeltery put an export bus on the controller to insert lead 1 ingot at a time. Then export obsidian to the casting Basin and import glass with a level emitter to keep say 20 or 50 on hand and walk away. Should be small enough to sink in a wall somewhere with a lava source to the tank and forget about it. Thanks a lot for all the answer. BC gates does not recognize casting bassin as an inventory. But I just found a solution ! With SFM for each trigger I : - Put an obsidian in the CB - Pour 72ml of Lead in the CB (with the Input/Output liquid) - Export Hardened Glass to the chest with an import bus - Empty casting bassin to the smeltery (with the Input/Output liquid) because sometime some liquid go to the bassin before the next trigger step Hope this could help somebody. There would be no problem if casting bassin was considered like a normal inventory though... That's why the setup I listed used the chest for the inventory to check, rather than the casting basin. Legacy Green Slime Crystal Slime Crystals are one of many materials used to craft and maintain tools in Tinker's Construct. It is a non-metallic material that can be worked directly from the Part Builder. Using Slime Crystals to make tool parts gives tools the Slimy ability, giving it greatly increased durability And the ability to randomly spawn small blue slimes when fighting. Crafting[] A Slime Crystal is made by smelting Slimy Mud in a Furnace. The resulting crystal will be either blue or green, depending on if the player used Gelatinous Slime or vanilla Slimeballs. Green Slime Crystal Usage[] Slime Crystals can be used in the Part Builder, making any part from a pattern. Slimy tool parts have high durability, low reinforcement, high handle modifier, and good speed. However, this comes at the cost of not being able to mine higher-tier materials. However, because of blue slime's excellent handle modifier (Best for any non-metallic material, and the second best handle modifier overall, at par with Ardite and just below Manyullyn) it is commonly used for tool handles. Both blue and green slime tool parts can be found in village tinkerers. Heya! Imma looking for a way to disable a recipe from the induction smelter from Thermal Expansion. Turns out it allows one to craft refined iron with barely any effort, compared to the actual intended way of having to jump through several IC2 hoops to get it, in the current version included in the DW20 for MC 1.7.1. I browsed through the configs a couple times but couldn't find anything for it. Neither in the IC2 config. Any ideas? I looked through some of the instructions for it, and it showed me how to tweak shaped and shapeless recipes, as well as furnace recipes, and then purge recipes even from other mods, but it seems disabling something for a specific machine from another mod was out of the scope of the instructions, OR I didn't pay enough attention, but in any case, seems very complicated unless someone can walk me through the script for an induction smelter recipe? EDIT: Looking some instructions from youtube, this is as far as I got: mods.thermalexpansion.Smelter.removeRecipe(); However the syntax is wrong and I get errors all around. Last edited: Jul 30, 2015 I looked through some of the instructions for it, and it showed me how to tweak shaped and shapeless recipes, as well as furnace recipes, and then purge recipes even from other mods, but it seems disabling something for a specific machine from another mod was out of the scope of the instructions, OR I didn't pay enough attention, but in any case, seems very complicated unless someone can walk me through the script for an induction smelter recipe? EDIT: Looking some instructions from youtube, this is as far as I got: mods.thermalexpansion.Smelter.removeRecipe(); However the syntax is wrong and I get errors all around. your minecraft folder > config\cofh\thermalexpansion > open common.cfg with notepad and change "B:Steel" to "false" Steel is the same as refined iron in this instance? Because I got no beef with steel. Only the easy method of making refined iron. You'll have to disable Steel making in the Induction Smelter. There's a direct config option for that. The issue is that IC2 considers refined iron to be steel. Welp. You'll have to disable Steel making in the Induction Smelter. There's a direct config option for that. The issue is that IC2 considers refined iron to be steel. Welp. Thanks! That solves my problem neatly then Hey lemming, is your config for dynamo base energy: 80, the RF output per tick the dynamos put out, or something else? I'm trying to tune it down but changing it to 50 didn't seem to change the rf per tick on the corresponding dynamo. Thanks! That solves my problem neatly then Hey lemming, is your config for dynamo base energy: 80, the RF output per tick the dynamos put out, or something else? I'm trying to tune it down but changing it to 50 didn't seem to change the rf per tick on the corresponding dynamo. Honestly not sure what to tell you. It works fine for me (just tested), and presumably everyone else as well. It's in common.cfg, under Dynamo, subcategory . Honestly not sure what to tell you. It works fine for me (just tested), and presumably everyone else as well. It's in common.cfg, under Dynamo, subcategory . So weird. As you see, I disabled the enervation dynamo, and while it's still in NEI, it can't actually be crafted. However, RF/t is the default 80 RF/t in-game. So weird. As you see, I disabled the enervation dynamo, and while it's still in NEI, it can't actually be crafted. However, RF/t is the default 80 RF/t in-game. i recommend that you update all of CoFH mods to the latest version and delete all config files then edit the new ones also when you disable something from thermal expansion it won't disappear from NEI only the recipe is disabled so that's normal , if you want to hide a specific item then use Inpure Core This article is in need of a cleanup. You can help Tinkers' Construct Wiki by organizing the information on this page. This article is a stub. You can help Tinkers' Construct Wiki by expanding it. Slimesteel is an alloy created by combining Molten Iron, Skyslime, and Seared Stone in a smeltery, at a 1:1:1 ration, yielding two ingots. It can be used for tools, and is a 1.16.5 exclusive. It has a mining level equal to iron tools, and applies the Overslime material trait to tools. The Overslime trait allows for slime crystals to be used to add Overslime to a tool, which adds a durability buffer. Overslime can be refilled by applying slime crystals to the tool in a Tinker's Anvil or Tinker Station. Applying a Slime Crystal adds 10 Overslime, a Skyslime Crystal 40, Ichor Crystal 100, and Enderslime Crystal 200. The Overslime trait allows 200 Overslime to be applied. Slimesteel parts also have one other trait, Overcast. Overcast allows 75 more Overslime to be applied per level of Overcast. {{stub}} Skip to main content Reddit and its partners use cookies and similar technologies to provide you with a better experience. By accepting all cookies, you agree to our use of cookies to deliver and maintain our services and site, improve the quality of Reddit, personalize Reddit content and advertising, and measure the effectiveness of advertising. By rejecting non-essential cookies, Reddit may still use certain cookies to ensure the proper functionality of our platform. For more information, please see our Cookie Notice and our Privacy Policy. Ok I found a slime island in the sky and collected enough slimes to make a nice stack of slime crystals. The problem is that they don't seem to work in any of the recipes in either the Tool Station or the Tool Forge. Any help would be appreciated. Maybe a link that shows how to use them? Ok I found a slime island in the sky and collected enough slimes to make a nice stack of slime crystals. The problem is that they don't seem to work in any of the recipes in either the Tool Station or the Tool Forge. 1 slime crystals are used in part builder 2 make sure you're not confusing slime balls with slime crystals Last edited: Mar 16, 2014 Reactions: ThatOneSlowking You smelted them, right? Assuming you did as you called them slime crystals, rather than slime balls, but just checking. To make Slime Boots in Tinkers Construct, arrange 2 Slime Balls and 2 Congealed Slime Blocks in a boot pattern. Smelt Slime Mud in a furnace to create Slime Crystals for crafting. Slime Boots prevent fall damage and enhance travel, making them perfect for adventurous gameplay. The bounce feature enables players to jump higher from falls, negating fall damage effectively. This is useful in exploring high terrain or avoiding deadly drops. Moreover, Slime Boots also provide a lightweight feeling, improving mobility in various environments. Once you have crafted Slime Boots, you can upgrade them further for enhanced effects. For instance, you might explore the addition of modifiers such as additional durability or speed boosts. Enhancements can maximize the potential of your boots. In the next section, we will delve into how to combine Slime Boots with other Tinkers' tools and gear. We will explore synergy between different enhancements and discover how to create the ultimate set of equipment for your adventures. What Are Slime Boots and Their Significance in Tinkers' Construct? Slime boots are a type of equipment in Tinkers' Construct that grant players enhanced mobility and protection. They allow players to bounce when landing and provide negation of fall damage, making exploration and traversal easier. Key characteristics of slime boots include: 1. Bounce effect 2. Fall damage negation 3. Customizable attributes 4. Special material combinations 5. Comparisons with other armor types The significance of slime boots in Tinkers' Construct extends beyond just basic functionality. Bounce Effect: The bounce effect provided by slime boots allows players to jump higher and travel further when landing on the ground. This feature can help in navigating difficult terrain and avoiding obstacles. Fall Damage Negation: Slime boots prevent fall damage, making them essential for players who frequently explore tall structures or jump from great heights. This ability improves survivability significantly. Customizable Attributes: Players can customize slime boots by adding materials and modifiers, which can enhance their performance. This customization allows for tailored gameplay experiences. Special Material Combinations: Different materials can be combined to create unique versions of slime boots, each with particular strengths. For example, using different slime types can yield varying bounce heights or resistance levels. Comparisons with Other Armor Types: Compared to standard armor sets, slime boots offer unique advantages but lack the overall protection provided by other armor. Players often choose between defense and mobility based on personal gameplay style. In conclusion, slime boots serve a multifaceted role in Tinkers' Construct, appealing to both mobility-focused players and those who prioritize survival during exploration. What Materials Do You Need to Craft Slime Boots in Tinkers' Construct? To craft Slime Boots in Tinkers' Construct, you need the following materials: Green Slime Jelly Blue Slime Jelly A boot pattern These materials come together to create boots that provide unique benefits, such as bounce and damage negation. Understanding the crafting process may lead to different attribute combinations for enhanced performance. Green Slime Jelly: Green Slime Jelly is a material obtained from Green Slime. When the slime is killed, it drops jelly. This jelly provides a bounce effect that allows players to jump higher when equipped. It is also relatively easier to obtain, as Green Slimes are common in specific biomes, such as swamps. Blue Slime Jelly: Blue Slime Jelly is derived from Blue Slime, and it provides an additional bounce effect as well as increased damage negation compared to the Green Slime Jelly. Blue Slime is less common than Green Slime and tends to spawn in higher elevations. Acquiring Blue Slime Jelly can enhance the functionality of the Slime Boots significantly. Boot Pattern: The boot pattern is a template needed for crafting in Tinkers' Construct. Players can create this pattern using wood or other materials, and it is essential for shaping the Slime Boots. The pattern can be customized to influence the properties of the crafted boots. In summary, crafting Slime Boots with various materials contributes to their effects and functionality in gameplay, enhancing the player experience. Which Types of Slime Contribute to Crafting Slime Boots? The types of slime that contribute to crafting slime boots are as follows: Green Slime Blue Slime Purple Slime The variations in slime types lead to different attributes when making slime boots. Understanding these variations helps in creating boots that best suit your gameplay. Green Slime: Green slime is used to create slime boots with excellent bounce attributes. This type of slime is commonly found in the Tinkers' Construct mod and provides cushioning for landing, which can negate fall damage. The bounce feature is particularly useful in exploration and combat scenarios. Evidence from player experiences suggests that boots made with green slime allow players to jump higher and fall greater distances safely. Blue Slime: Blue slime contributes to creating slime boots that facilitate greater mobility. The boots made with this type of slime offer faster movement speeds. Players often prefer blue slime boots when navigating large areas or when speed is critical during avoidance or escape strategies. According to community feedback, blue slime boots enhance agility, making them favorable for quick traversals and dynamic gameplay. Purple Slime: Purple slime is used to craft slime boots that offer unique visual aesthetics while also having a balance between bounce and mobility. Although it may not specialize in either attribute as strongly as the green or blue slimes, it provides a well-rounded option. Players appreciate the look and feel of purple slime boots, often indicating that they combine function with style. In some cases, players favor purple slime boots for their versatility in various gameplay situations. What Other Resources Are Required for Making Slime Boots? To make slime boots in Tinkers' Construct, you need specific ingredients and resources. The main resources required for making slime boots include: 1. Slime crystals 2. Leather 3. Tinkers' construct crafting station 4. Tool station Transitioning to further details helps illustrate the versatile aspects of these resources. Slime Crystals: Slime crystals are obtained from slime balls. Slime balls are harvested from slimes found in specific biomes or created in slime blocks. Different colors of slimes yield different types of slime crystals, allowing for various attributes in the boots. Leather: Leather is crafted from animal hides, typically cows and horses. It is a fundamental resource in Tinkers' Construct, providing durability and additional attributes to the finished product. Tinkers' Construct Crafting Station: The crafting station is essential for creating Tinkers' tools and equipment. This station allows players to combine resources and see the different properties of the items they create. Tool Station: The tool station is necessary for assembling and modifying tools and materials. It enables the player to apply additional modifiers to the slime boots, enhancing their effectiveness. These resources enhance the functionality of slime boots, providing bounce and damage reduction. For example, using different colored slime crystals may alter the boots' bounciness and durability. Thus, the choice of resources can shape gameplay experiences uniquely. How Do You Craft Slime Boots in Tinkers' Construct? You can craft Slime Boots in Tinkers' Construct by using specific materials and a Tool Station. The process involves combining slime crystals and a material such as leather or a slime-related item. To craft Slime Boots, follow these steps: Gather Materials: You will need two Slime Balls or Slime Crystals and a crafting material like leather or a slime block. – Slime Balls can be obtained by killing Slimes or by melting Slime Blocks in a furnace. – Slime Crystals can be made by using a Tinkers' Construct Smeltery and creating a crystal from a slime liquid. Access the Tool Station: – The Tool Station allows you to craft and modify tools and armor in Tinkers' Construct. Ensure you have built one in your Minecraft world. Initiate Crafting Process: – Open the Tool Station interface and select the option to create armor. – Place the required materials in the crafting grid following the layout specific for boots (typically, the items are arranged in a boot shape). Complete the Crafting: – Once the materials are placed correctly, you will receive the Slime Boots in your inventory. – These boots provide bounce and negate fall damage, making them useful for exploration. Slime Boots enhance your gameplay experience by allowing players to traverse heights easily and avoid damage from falls. The crafting process is straightforward and provides a practical benefit in the game environment. What Is the Step-by-Step Process for Crafting Slime Boots? Slime boots are specialized footwear in the game Minecraft, particularly within the Tinkers' Construct mod, that allow players to bounce from heights and reduce fall damage. The boots are crafted using slime materials, providing unique gameplay advantages. According to the official Tinkers' Construct Wiki, slime boots grant players the ability to jump higher and negate fall damage through their unique properties. This enhances player mobility and exploration options in the game. To craft slime boots, players must gather specific resources. The primary materials include slime crystals, which can be obtained by defeating slime mobs or collecting slime blocks. Players also require a crafting station to combine the materials. In addition to the Tinkers' Construct Wiki, the Minecraft community forums often describe crafting details and mechanics. These sources emphasize the strategic advantages of using slime-based items to gain mobility. Players can be motivated to craft slime boots due to the benefits they offer in parkour challenges and exploration scenarios. This heightened mobility can help players access hard-to-reach areas and avoid fall damage. The enjoyment of enhanced movement mechanics in gaming can lead to increased player engagement. A study published by the International Journal of Gaming and Computer-Mediated Simulations shows that custom items can improve overall user satisfaction. Slime boots create opportunities for creative gameplay and strategy. Players can utilize them in various scenarios, thus enriching their gaming experience and fostering deeper player interactions. To craft slime boots, players should focus on collecting slime-related resources and utilizing available crafting stations efficiently. Engaging tutorials on platform sites like YouTube can help players with detailed crafting steps. Potential strategies include collaborating with other players to gather materials faster, or trading for slime crystals. Leveraging community resources enhances the crafting experience and accelerates resource acquisition. Ultimately, slime boots represent an engaging aspect of Minecraft gameplay, enabling players to explore and navigate their worlds in innovative ways. What Are the Benefits of Using Slime Boots in Tinkers' Construct? Slime boots in Tinkers' Construct offer players several practical benefits in gameplay. Bouncy effect Fall damage negation Terrain traversal enhancement Crafting versatility Customizable attributes Advantages in combat situations These benefits demonstrate the practical uses of slime boots, enhancing both mobility and survival in the game. Bouncy Effect: The bouncy effect of slime boots allows players to jump higher and land safely from great heights. This feature provides significant vertical mobility, enabling players to reach elevated locations easily. In gameplay scenarios, this can help players avoid obstacles or navigate challenging environments quickly. Fall Damage Negation: Slime boots negate fall damage, providing a major advantage when exploring or traversing landscapes. Players can drop from heights without suffering injury, which encourages a more daring style of exploration. For instance, players can scope out locations from high vantage points without the fear of severe damage. Terrain Traversal Enhancement: Slime boots contribute to improved movement across various terrains. Players can navigate through difficult landscapes, such as swamps or slime blocks, without being hindered by slow movement. This feature offers a strategic advantage when exploring or fleeing from enemies. Crafting Versatility: The crafting process allows players to customize slime boots with different materials and modifiers. This adaptability leads to various combinations that cater to individual gameplay styles. Players can prioritize attributes like durability or additional jump height, tailoring