How does minecraft towny work

Towny Towny is a great plugin used on elected officials. Towny is a director of the resident-city-nation hierarchy. It also provides a block authorization manager for a network-like protection system that prevents grief. The networks are based on Minecraft chunks which are 16x16 plots. Every city has a mayor and the mayor has the option to create a nation and be king. A mayor and a king can create assistants to help them run their city, or their nation. Starting a city to start the fun you will need to put in some time and effort. It costs 20,000 to create a new city. The best method to make this money is to go to waste, mining, and then sell many of the boxes you find. Once you have enough money you need to explore the desert and find the area you want to create your new city B. After you have found a nice area, a new type/city (city name). Now you are the proud owner of your new city if you want to add city assistants you can type/rank city add (playername) helping your city expansion at first the city you create will be only 4 plots. In order to expand your city you will need to invite more players and increase the residents living in your city. You get 4 new plots per resident you add. In order to claim new plots, meet them and type/claim in the city if you have difficulty distingu distingu distingu distingu plot boundaries you can type/map a town which will help with plot guidelines and another command is/res to go through the border plot which will create a puff of smoke depicting a quick plot and a way of connecting to get a lot of plots is to join the nation! Joining the nation will give the city a maximum of 280 bonus plots. New resident plots will be looking for plots to build on. A plot is a 16x16 chunk of rock to the sky that will protect everything that builds within its boundaries. When loading new plots into your city, they cost $1400. Once you've made a new conspiracy, you can't drop the claim. You'll need to get enough money in your city bank to claim new plots. You can deposit money into the bank by typing/depositing in the city (amount). Mayors can withdraw the money by typing/city withdraw (amount). To eject a plot for a sale type/plot fs (amount) to have a resident claim the plot, they must meet it and type/plot claim residents can allow their friends to build and destroy on their pitch by typing/res a member to add (playername) you can also sell plots to players who live in other cities by setting up your pitch as an embassy. To do this type/plot set up embassy if a resident leaves town and leaves items in their plot, you can ask a staff member to remove their private chest, and the rest stays for the mayor. Outposts If you want to claim an area far from your city, you can do so by claiming it as an outpost. To require an outpost of type/outpost claim the outpost will cost 60k to claim. If residents want to beam to the outpost, they should type/city Nations If you feel you have achieved everything within your city, you can always start a nation! Buying a nation costs 300,000. It also has $350 daily maintenance. Money needs to be in your nation's bank to pay for daily maintenance. You deposit money by typing/depositing the nation (amount) to withdraw money from a nation bank you can type/nation withdraw (amount) as king you can add new cities to your nation by typing/nation add city (city name) more cities will add, more bonus plots your nation will give. The maximum plot bonus is 280 plots. Towny is a great plugin that protects construction as well as a strong community saddle. If you need help with something to do with towny just ask a staff member and they will be happy to help you. Skip to content Use this page to learn how Towny works, how different settings affect the game, what you can do to customize Towny to your liking. Hierarchy nomads are simply players who are not part of any city. They are landless and their permission nodes can be configured using TownyPerms.yml. Nomads can purchase embassy plots if they are given towny.command.plot.claim in Townyperms.yml residents any person who joins your server can be a resident of a city, (by default they get towny.town.resident node permission in the nomadic section of townyperms.yml.) residents have their own command/resident who is used by himself to output a resident status screen, displaying money, city, owned plots and friends. Residents can join cities or choose to start their own city. You can also automatically check residents into one city when they join the server for the first time by setting the default_town_name: '' in config.yml. Residents joining cities can claim a scheme the mayor has set up for sale. When a resident has one or more plots, they will see a new line on their Resident screen, showing owned plots and a default Prebble line showing the plot stalars given in all plots owned by the resident (who have no custom plot line set.) Residents have their credential nodes configurable using TownyPerms.yml. City towns are a collection of residents (or just one resident) with one resident as mayor. The city also has a bank that the mayor can withdraw from. A mayor can also be aides who have the same powers as his/herself. Cities can receive taxes to be taken at the end of each day interval. Towns usually grow outward from their block, the obstruction the mayor met during the city's founding. The blocks should be claimed near other checkpoints unless the mayor claims a wilderness outpost through a sea outpost. Cities can be restricted to multiple residents by using global_town_settings.max_residents_per_town configuration option, by default this is not limited. You can see all cities on the server in the list by using /List of Cities. As of city 0.95.1.0 you can do this Town metadata, see here for details. Mayors run cities and work with the help of their assistants, city managers and residents. Mayors have their credential nodes configurable using TownyPerms.yml. Mayors can decide which rank their residents, in their city. This can be a city assistant or any other custom ranks created by the server administrator in the townyperms.yml file. Mayors can see the rows available by using the /town ranklist command. Players rated with a /town {add|remove} {playername} {rankname} rating. An actor can have more than one degree assigned, allowing managers to create diverse city roles such as bankers, builders, inviting the mayor to choose for their trusted residents. Mayors also determine what kind of tax rates and city tax will be charged to residents. You can't run two cities unless you're also an admin. An administrator can do the following to manage two or more cities: Example: Admin Bob/ How to Create NPC Towns. Director Bob wants him to have a server city, and his city. Bob will start by creating his server city and setting taxes, narrative prices, permissions. This type of city should not give residents, allies or foreigners permissions in the server city. Bob can give himself more city blocks using the /ta givebonus {townname} command {#}. When Bob finishes turning his city the way he wants he uses/townyadmin set mayor {townname} npc to place a fake resident 'npc' as mayor of the server city. This is usually enough to protect the sludge zones of most servers. Then Bob leaves server town and creates his own city. Using mayor/townyadmin set {townname} npc command Bob can flip back and forth between cities. Bob doesn't have to leave his city to add players to server city though! He can use /townyadmin {townname} to add {playername} to add players to server city or set default_town_name: 'Server_Town' in config.yml. Bob can also add NPC to the nation using /ta nation {nation} to add {town}. As of Towny 0.95.1.0 mayors can define titles (prefix) and family names (postfixes) for their city residents. This is done with: /town set title {name} titlegoeshere /town set last name {name} surnamesgoeshere Outlaws As of Towny 0.92.0.0, cities (usually mayors by default, but maybe other city rows,) can define a list of outlaws. Outlaws are defined by using the /city outside the law [add/remove] [name] and the command requires a towny.command.town.outlaw permission node. Outlaws can be any player and don't have to be in a city or a country. If the new vigilante is a member of your town, they'll kick. Cities that have themselves set open status (anyone can join using the /town join command,) can use the outlaw list to prevent these players from joining their city freely. Players can't spawn into public cities that are as important to them as outlaws. Players who enter the city where they are Being an outer will see a warning-title message informing them. If a player is online and they have become a vigilante they will see a message in the chat. Vigilantes can be sent to prison if they die in a city where they are considered vigilantes. This is jail.is_jailing_attacking_outlaws this option in Towny Config.yml to be set to true. It also requires the person who did the murder to have towny.outlaw.jailer junction authority. It also requires the city to own at least one prison plot. By default only mayors, aides and sheriff ranks have the towny.outlaw.jailer permission node. A member of the city can view their city's vigilante list using the city's outlaw list. Anyone can see the outlawed list of any city using the /town outlawlist {townname} Nations A nation is a collection of towns (or just one city) with one capital city. The mayor of this capital is king. A nation can join the event of war, as well as an ally of other countries. The nation also has its own bank. He can also dissolve the cities that belong to him. Countries can also be a creep reached through/nation spawn which if the nation is considered 'public' can be reached by almost any non-enemy players. The nation's creeps can be moved to the capital at config.yml at global_nation_settings.capital_spawn, otherwise a spawning point could be anywhere in the country. Two countries can decide to join the alliance, which allows them to be protected from friendly fire, to help each other with plots (if the plot's forslnation line allows allies,) and help each other in war. Starting from 0.91.0.0, you can limit nation alliances to be 2-way only. So that Nation A cannot see Nation B as an ally unless Nation B also considers the nation an ally. You can enable this setting in the configuration: war.disallow_one_way_alliance, the default is false. Also, as of 0.91.0.0, you can limit who can create, join, and maintain a nation by requiring a minimum number of residents. See config.yml Global City Settings. From 0.92.0.0, you can set a maximum distance between the nation's capital and cities that are allowed to join the nation. See config.yml Global City Settings. As of 0.93.0.0, countries can provide NationZone encompassing its member cities. This is available at global_nation_settings.nationzone.enable at config.yml. The regions of the nation are just like a normal meddly, but the only players who can change the region are members of the nation. It can be useful to avoid greifing near cities that have a nation. NationZones can be increased in size by increasing the nation's population through NationLevels at config.yml, you can optionally make the capital city have a larger NationZone. NationZones can be disabled during wartime at config.yml global_nation_settings.nationzone.war_disables. Starting at 0.96.2.0 you can configure a maximum number of residents per city They have no nation, in global_town_settings.maximum_number_residents_without_nation. When set up over 0 cities will need to establish states to add more residents. States can grant many benefits to their towns which can increase as the nation's population grows, these include: bonus urban blocks to claim. Cheaper city maintenance costs. Bigger around the nation's regions. Increased limit of posts. Kings lead the nations and are the mayor of the capital. Kings have their configurable certification nodes through TownyPerms.yml. Kings can decide which ranks their residents fall, in their nation. It can be a nation helper or any other custom ranks created by the server administrator in the townyperms.yml file. Kings can see the rows available by using the /nation ranklist command. Players rated using the /nation rating {add|remove} {playername} {rankname}. An actor can have more than one degree assigned, allowing executives to create diverse nation roles such as bankers, inviting the king to choose for their trusted residents. Kings have the ability to define titles (prefix) and surnames (after efrims) for the residents of the cities they have in their nation. This is done with: /nation set title {name} titlegoeshere /nation set last name {name} Last name goes here typing the commands with nothing after the player's name resets the title or last name is empty. The Townyperms.yml configuration and the roles of the rows within Towny True 0.82.0.0 Towny have a permission configuration file, TownyPerms.yml, located in towny \settings\folder. This system pushes permissions directly to Bukit and works alongside all other alley plugins. It allows you to set sets of permissions based on player status (nomad/resident/mayor/king). You can also assign additional permissions based on all assigned city/nation ranks (assistant/VIP, etc.). This system is not restricted to Towny permission orders. You can assign permissions for all plugins in its groups. This file allows managers to decide what each player rating can do. Some ranks are automatically assigned: players without cities are nomads. Players in cities are residents. City owners are mayors. Nations are kings. Some ranks are assigned by mayors or kings, and complete the ranks that players already have: Mayors can become a city assistant resident. Kings can turn a resident into an assistant nation. Mayors and kings can award ranks created by an administrator, allowing for diverse customization. An actor can achieve many complementary ranks from their mayor or king, allowing for diverse nation/city roles. Examples of this will be town builders, municipal bankers, nation bankers, town invitees, etc. A city resident can see the ranks within their city using the city rating list. A mayor can use /town rank {add|remove} {playername} {rankname} to give a player a new rank within his city. King can use /nation rating {add|remove} {playername} Give a player a new rank within their nation. Configuring mayors and titles king, city and nation names Towny gives you the ability to customize the naming program that applies to mayors, kings, cities, capital cities and states. This is done with two sections in town_level and nation_level town_level configuration 1: The basic layout of the townLevel lines are as follows: - numResidents: 1 namePrefix: '' namePostfix: ' (Settlement)' Mayor Seamstress: 'Monk' MayorPost: '' CityBlockBuyBonusLimit: 0 townBlockLimit: 16 Modular maintenance: 1.0 townOutPostLimit: 0 debtCapModifier: 1.0 - Residents: 2 NameFrafax: '' Post Name: ' (Hamlet)' MayorPrefix: 'Chief' MayorPostfix: '' CityBlockBuyBonusLimit: 0 townBlockLimit: 3 2 MaintenanceModifier: 1.0 townOutpostLimit: 1 debtCapModifier: 1.0 These are called as follows: Variable description numresidents: 1 This is how many residents in the city need to get to town_level. namePrefix: '' This value is added to the front of the city name. namePostfix: ' (settlement)' plugin for the end of the city name. Mayor Fairfax: This monk is added to the front of the mayor's name. Mayor Post-Flex: '' It's added to the end of the mayor's name. townBlockBuyBonusLimit: 0 This is the maximum number of plots which the city can buy through/city buy bonus. Requires town.max_purchased_blocks_uses_town_levels is set to True in config.yml. townBlockLimit: 16 It bypasses the town_block_ratio and decides how many city blocks the city gets. Requires town_block_ratio: '0' defined in config.yml. MaintenanceModifier: 1.0 Use a higher multiplier to increase maintenance as cities get more residents. Does not affect servers with town_plotbased_upkeep:true in this case it is based on plot count and not resident count, unless you also have town_plotbased_upkeep_affected_by_town_level_modifier:true) townOutpostLimit: 1 is how many outposts the city can claim. Requires limit_outposts_using_town_and_nation_levels is set to True in config.yml. debtCapModifier: 1.0 When debt_cap_uses_town_levels is set to correct config.yml, debt_cap override price will be doubled by debtCapModifier town_level. (Formerly: DebtCapModifier of 3.0 and debt_cap.Overtaking of 1000.0 will set debt of 3.0 x 1000 = 3000.) The two levels above are for cities of 1 and 2 inhabitants. When a city is created the new city of the mayor has (a settlement) added to the end of his city name and he gets the prefix of a monk. When the mayor gets a resident of both his town becomes named the city (Hamlet) and he gets the prefix of chief. nation_level: The basic layout of the NationLevel lines are as follows: - numResidents: 10 capitalPrefix: '' capitalPostfix: '' namePrefix: 'Federation of ' namePostfix: ' (Nation)' ' KingPrefix: 'Counting ' kingPostfix: '' CityBlockLimitBonus: 10 MaintenanceModifier: 1.0 NationTownUpkeepModifier: 1.0 NationZonesSSize: 1 2 - numResidents: 20 capitalPrefix: '' capitalPostfix: '' NamePrefix: 'Dominwon's ' NamePost: ' (Nation)' King translated: 'Duke' kingPostfix: '' CityBlockLimitBonus: 20 Maintenance Module: 1.0 NationTownUpkeepModifier: 1.0 nationZonesSSize: 2 nationBonusOutpostLimit: 3 These are named as follows: NumResidents Variable Description: 10 This is the number of nation residents must reach the nation level. capitalPrefix: '' It is added to the front of the nation's capital city. capitolPostfix: '' It is added to the end of the nation's capital city.

NamePrefix: 'Federation of ' it was added to the forefront of the nation's name. namePostfix: ' (Nation)' It was added to the end of the nation's name. King Perpicex: This 'Count' is added to the forefront of the nation's leader. kingPostfix: '' It is added to the end of the name of the nation's leader. townBlockLimitBonus: 10 This is the number of blockages of bonus city given to the city when they join the nation. MaintenanceModifier: 1.0 Use a higher multiplier to increase maintenance as states get more residents (unless you use town_plotbased_upkeep:True in this case it is based on a plot count rather than a resident count.) nationTownUpkeepModifier: 1.0 Joining the nation will be low/raise how much maintenance your city pays. This multiplier is calculated after all other multipliers have an impact on the city's maintenance cost. When set to 1.0, there is no change in maintenance. NationZonesSSize: 2 several wide plots nationZone is around a city. NationBonusOutpostLimit: 2 few additional outposts the city can claim on top of its normal limit. How cities grow Mayors start cities by using the new /city command {townname}. This will often require an amount of money set up in a price_new_town. The block they're standing in will be the city's home block, the exact place/position will be the city's point. A mayor can move the point of creeps inside the blockage using the creeps. The hold can be moved to another city block loaded using /t set homeblock. Additional city blocks can be claimed through the city lawsuit. These blocks should be directly adjacent to the already claimed blocks, unless it's an outpost. Joining cities there are two ways to join cities, the first being by being invited by a mayor or an aide in the city. The second is by joining an open city. Mayors and assistants can add players to their city by using the /town add {playername} command. The player will receive a request to accept or decline the invitation. Mayors can set their cities to open using/city to move open. A player who is not in the city can already use the /city command to join {townname} to join open cities. Open towns can be viewed using the /town list by opening. When residents join cities, they increase the number of obstructions accessible to the mayor, claiming. Towny land ownership plotting system provides manager with hands-free operation server To block protection. Block protection is divided into earthy plots, called urban blocks, which by default are 16x16x256 (the full height of the world.) will think of them as cells in a uniform grid, all aligned and no space between each other, each the same size. Urban blocks are loaded by mayors who can then sell/give individual plots to their city residents. As of Towny 0.95.1.0 MetaData can be stored in a city block, see here for details. City block size you resize config.yml city block in town_block_size:16. This value change is suggested only when you first install Towny. We'll do this after entering data, you move things around unwantedly. Using a smaller value will allow for higher accuracy, at the cost of setting up more work. Also, very small values will make the water storage system used by Towny useless (not recommended to drop below 4.) each cell is (town_block_size x town_block_size x 256) in size, with 256 being rock to clouds. Claiming towns residents can claim the city blocks as long as they have the right certification binges. By default, this is limited to mayors and people with an assistant rating. The player uses /city claim or/city claim #or/city to sue rect|circ #or/city to sue auto to claim city blocks for their city. /Town claim - Load one pitch, where the player stands. /City Claim # - Load a square with a radius equal to #נתון. / City claim rect # - Load a square with a radius worth #נתון. /City claims circ #- Load a circle with a radius equal to #נתון. /City Auto Load - Load all possible blocks in the city squarely. Using the /town command will save some available claim holds. Starting from 0.95.0.0 you can set a refund amount for unanswered city blocks in economy.new_expand.price_claim_townblock_refund, we do not recommend that you set this amount or higher than the cost to claim a city block. Starting from 0.95.0.0 you can make the cost of claiming city blocks increase by economy.new_expand.price_claim_townblock_increase. When set to 1, this session is disabled. 1.3 means +30% per bonus claim block cost. You can see the cost increase in price/towny output. Starting at 0.96.3.0 you can set the maximum price that can be reached economy.new_expand.price_claim_townblock_increase. This configuration is economy.new_expand.max_price_claim_townblock in config.yml. Setting up how many city streets the city gets you can change how many streets the city gets to claim. It's done in two places. The town first checks config.yml for town_block_ratio: 8 and default gives the city 8 blocks per resident. You can work around this by using town_block_ratio: 0 and using the town_level section of config.yml More information about the town_level and how to configure it is here. Buying urban blocks Mayors can buy city blocks using Buy bonus {amount}. There are two methods that you can limit the number of holds that can be purchased by a town: if town.max_purchased_blocks_uses_town_levels is set falsely: the maximum is set in config.yml by max_purchased_blocks: '0'. If town.max_purchased_blocks_uses_town_levels is defined as true: the maximum is dictated through town_level town_level's BlockBuyBonusLimit, allowing cities with more residents to purchase more city blocks. The price of a city block bought is also set at config.yml at price_purchased_bonus_townblock: '25.0'. The price can be set to increase with each purchase using price_purchased_bonus_townblock_increase: Configure '1.0'. With this feature, mayors can grow their city without the need for new residents. Increasing costs can be seen/city buy bonus output. As of Towny 0.96.3.0, you can set a price price_purchased_bonus_townblock_increase the plan can switch to. An unsavory economy.new_expand will result in vibrant lively lively and vibrant bearings can be grouped plots of plotting groups together into plot groups. To create a group, the Player must have the appropriate permissions to run on it. The player then meets the plot and performs: /plot group add {group_name}, Towny will add the plot you are standing in, to the group if it exists, or create a new group if a group with that name does not exist. Just repeat the /Word Plot Set Insert {same_group_as_above} command when standing in other plots to add these plots to the group. Most other commands used to handle the same debasses except for one exception, instead of starting with the prefix /plot, plot group operations begin with the prefix /plot group. For example, to set up a property for sale you are making /plot fs 10000, but for a plot group called highlands you would make /plot group fs 10000. Smets, types and others follow that convention. See /Plot group? For a complete list of commands. Player pitch types can use the city /city plots {townname} to display the counts of different label types in the city. Towny post-0.75 added plot types except the default. It's to give mayors more control over their cities. As of Towny 0.95.1.0, you can configure costs to define plot_type_costs in config.yml. The plot types are as follows: These smooth default labels do not need any specific command to be defined. They are for sale with /plot forsale {$$}. Nondefined plot can be reset to the default plot with /plot set reset. Shop lots are designed with a/plot to set up shop. A mayor can use the price of a city group store {$$} to define how many store plots are sold by default. This can be circumvented when a mayor puts the actual lot up for sale with /plot forsale {$$}. A mayor can also collect an additional store tax with a city set store tax {$$}. This tax is charged in addition to this tax A thy way. Arena plots plots plots designated plots with/plot set arena. The PVP is about constantly in arena plots as well as friendly fire damage. The city's health organization is also down in the arena plots. Embassy plots are designated with/plot to set up embassy. A mayor can use the embassy price of /town set {$$} to define how many embassy plots are sold by default. This can be circumvented when a mayor puts the actual lot up for sale with /plot forsale {$$}. A mayor can also collect an additional embassy tax with city embassy tax {$$}. This tax is charged in addition to the normal outline. An embassy plot can be bought by any player, whether they are in town or not, as long as they have the towny.command.plot.claim permission node. The block remains owned by the host town and a mayor from the host city can take the lot from the owner at any time. Embassy plots can also be changed to shop plots, allowing larger store cities, where many different players of cities can set up shops. When a player leaves town they do not lose ownership of their plots if these plots are set to be embassy plots. Wilds Wilds plots are designed with/plot set wilds. A wild plot allows residents to destroy the blocks found on the identification list to be ignored in the wild. This includes flower trees, flowers, mushrooms and other harvestable blocks by default. It does not include stone, dirt, grass and other terrain blocks. It is useful for creating a tree farm, and protecting the area around the city, while still allowing residents to cut trees and explore caves. Setting up wild plots can be a little complex, here are instructions. Navigate to your towny\data\worlds\WORLDNAME.txt file set up: unconscoveredZoneBuild= false without requiringZoneDestroy=false Configuring the uns suedZoneIgnoreIds line to include the blocks you want players to disconnect/build. Go to wilds courts you can set using/plot set wilds by default residents will build/destroy enabled for them, you can also set up perverse allies or foreigners if you want non-city members to use Wilds storylines. Inn Plots Inn's inn plots are for inn/plot set. Inns are most useful when deny_bed_use is correct on config.yml. This setting makes it so that players cannot use beds in plots they do not own personally, except where that plot is part of the Inn. The inn plot allows anyone to use the bed to set their creeps/res and spawn on death spots. However, The Inn Plots cannot be used by enemies of your nation. Prison plots plots are defined with/plots set up as prison. Players can be imprisoned if: the mayor/sheriffs of the player send them to prison using the /t pass prison [jailnumber] [residentname] {days} command. An assailant who attacks a town that considers him an enemy (nationality relations) has died in this city. He was sent to the magna city's first available prison plot. Attacker who attacks town who considers him a dead outer City by player with cityy.outlaw.jailer authority junction. He was sent to the magna city's first available prison plot. The jail.is_jailing_attacking_outlaws must be true. Incercerated actors become incarcerated if: they leave their city and become a nomad, the mayor/sheriff unjails them, the actor pays bail to the city that imprisoned them, (via: / paybail prison resident) they manage to escape the prison plot and the city and enter the wilderness. Plus: incarcerated players can't beam. Incarcerated players cannot use Ender Pearls unless they are activated in a configuration. Jailed actors who die are sent back to their scheduled prison plot. Incarcerated actors don't make financial payments when they're killed. Imprisoned players show their incarcerated status on the /res [playername] screen, along with the city in which they are imprisoned. He suggested you make a new rank in the townyperms.yml called Sheriff, and give this rank towny.command.town.toggle.jail intersection. New files created in townyperms.yml will contain this rating by default. There is a list configured in jail.blacklisted_commands you can define a list of commands that incarcerated players cannot use. Farm plots and farm plots are designed with a/plot to set up a farm. Farm court players only build/destroy blocks designed for config.yml Towny in global_town_settings.farm_plot_allow_blocks. By default this list includes: BAMBOO,BAMBOO_SAPLING,JUNGLE_LOG,JUNGLE_SAPLING,JUNGLE_LEAVES,OAK_LOG,OAK_SAPLING,OAK_LEAVES,BIRCH_LOG,BIRCH_SAPLING,BIRCH_LEAVES,ACACIA_LOG,ACACIA_SAPLING,ACACIA_LEAVES,DARK_OAK_LOG,DARK_OAK_SAPLING,DARK_OAK_LEAVES,SPRUCE_LOG,SPRUCE_SAPLING,SPRUCE_LEAVES, Beets, cocoa, CHORUS_PLANT,CHORUS_FLOWER,SWEET_BERRY_BUSH, seaweed, seagrass, TALL_SEAGRASS, grass, TALL_GRASS LARGE_FERN, shedding, carrots, wheat, potatoes, pumpkin, PUMPKIN_STEM,ATTACHED_PUMPKIN_STEM,NETHER_WART, cocoa, Vine, Hotel, MELON_STEM,ATTACHED_MELON_STEM,SUGAR_CANE, Cactus, Allium, AZURE_BLUET,BLUE_ORCHID, Dandelion Cornflour, Lilac, LILY_OF_THE_VALLEY,ORANGE_TULIP,OXEYE_DAISY, Lord, PINK_TULIP, Poppy, RED_TULIP,ROSE_BUSH, Sunflowers, WHITE_TULIP, Kemer, Rose , crimson, mushrooms, Crimson CRIMSON_HYPHAE,CRIMSON_ROOTS,MUSHROOM_STEM,NETHER_WART_BLOCK,BROWN_MUSHROOM,BROWN_MUSHROOM_BLOCK RED_MUSHROOM,RED_MUSHROOM_BLOCK,WARPED_FUNGUS,WARPED_HYPHAE,WARPED_ROOTS,WARPED_STEM,WARPED_WART_BLOCK,WEEPING_VINES_PLANT,WEEPING_VINES,NETHER_SPROUTS who can build/destroy these blocks is still determined by the twering line of the plot that appears to be on the plot screen. This means that if B = rnao, anyone can plant/place the blocks that are allowed in the plot. If B = r--- then only city dwellers can plant/place the permitted blocks. If managers want to, they can add FARMLAND to the list of allowed blocks, which will allow anyone allowed through The Perm Line to also make farmland with a hoe. By default, FARMLAND is not included and only administrators/mayors/assistants will be able to create With a hoe. The town is already protecting farmland from being cocked to the grass, so farmland will only return to the dirt if it is not launched. Farm plots also allow an actor to kill animals in a plot. In order to kill the list of defined animals the player must be able to break wheat blocks on the plot. The animal list is configured in global_town_settings.farm_animals. By default this list includes pork, cow, chicken, sheep, MOOSHROOM. Bank plots are designed with/plot to set up a bank. Bank plots can be used to restrict bank deposit/withdrawal in the city and country. By default this setting is turned off, but can be turned on in bank.is_banking_limited_to_bank_plots in more useful config.yml Bank plots for other plugins that will want to limit their banking features within Towny Bank plots. Outposts Are usually urban barriers around the block of the house, which are always connected to the other blocks of the city. To require a block in a wilderness or other world, a mayor or aide must require an outpost. In order for players to require outposts, the configuration must be configured as allow_outposts: true and players require towny.town.claim.outpost in their permission node group. You can view the list of postes by using the /town outpost list command. The scratch point can be moved inside an outpost block by using the /t set outpost command. City dwellers can beam to outposts to use the outpost /city x|name|name:x with x being number 1 - however many outposts there are in the city where the plot name is the plot name of the outpost (see in /t Outpost List and set using /plot set name {name}.) where name:1asdasda - useful when the plot name starts with a number. An administrator can configure the number of outposts a player can load, this is defined in the Information/Option/Meta-Node section of your Authorization Plugin by using the towny_maxoutposts: {number} info node. See here. Or, starting at 0.93.0.0, you can use config.yml's town_level and nation_level maxoutpost values to give larger cities outposts to claim, and bonus outposts for cities in countries. This requires that you enable the feature in config.yml in global_town_settings.limit_outposts_using_town_and_nation_levels. A manager can also set up the number of residents required by a town before they can claim outposts, using minimum_amount_of_residents_in_town_for_outpost in a way. There is no resident requirement by default. Outposts may not be too close to other blocks, just like when a mayor starts a city they may not be too close. The exact number is set in config.yml at min_distance_from_town_homeblock:5. You can also configure how close an outpost can get to any other city's schemes by using town.min_distance_for_outpost_from_plot. You can also set the cost of claiming an outpost in config.yml in 500.0. Land sale is sold by mayors to residents who are part of their city. using_economy: True must be set in config.yml to apply costs. Mayors have an order used in the game to set the cost of all pitches set for sale from now on. /town set plotprice {$} This defines the cost of new plots set for sale, plots already set for sale keep their costs. If it is not set, the labels will cost $0 by default. To put a land for sale to a mayor, when standing in the plot, type '/Plot for sale for {optional cost}'. The resident would then type '/plot claim' (while standing partly,) to buy it. Using map maps in the city shows the network system of plots. You can view the map once using /towny map or set the map to display each time you move from one block to another: use a toggle/resident map to turn it on or off. A larger version of the map can be seen using the large city map. Regeneration Plot &amp; Uncontored Plots There are 4 options to influence blocks/plots. Returning unclaimed city blocks to their original state on the unclaim when the city plot is unclaimed (by a player using /t to cancel the noon or through maintenance) it will slowly begin to return to pre-city mode. Blocks will slowly change back to the picture taken of the plot when it was first sued. A city block must return completely before the image of the block is removed. If the hold is returned halfway, no new image will be taken, and if the hold is not required again, it will return to the original image. Disabbing this feature is done in towny\data\worlds\worldname.txt @usingPlotManagementRevert=false, or by using /tw replace again while meeting the world in which you want to move it. Disabbing this feature for new worlds is configured in new_world_settings.plot_management.revert_on_unclaim.enabled. You can configure certain block types that you don't want to restore to prevent players from exploiting Regen for diamond boxes. Block types to be not restored are defined in plotManagementIgnoreIds in the world txt file city\data\worlds\worldname.txt. Defaults for new worlds are configured in new_world_settings.plot_management.revert_on_unclaim.block_ignore Deleting predefined blocks in unclaim when a city plot is not required (by a player using /t unclaim or through maintenance) block materials that match the list will be deleted within that block. This can be useful for deleting all signs within a block, ensuring all boxes locked with locket or Deadbolt signs are unlocked. Disabbing this feature is done in towny\data\worlds\worldname.txt @ usingPlotManagement deletelete=false Enabling this feature for new worlds is configured in new_world_settings.plot_management.block_delete.enabled You can configure the list of material names to be removed on a world-by-world basis. Materials listed in @plotManagementDeleting= will be removed from the block. Defaults for new worlds are configured in new_world_settings.plot_management.block_delete.unclaim_delete Plot-Owners and Mayors' /plot clear command feature is available only to mayors on public city soil: /plot clear. This ordinance was intended to be used after some of it was personally owned by a resident, who moved to another plot or left the city. By default this list includes only signs, useful for mayors to remove protection of signs on doors, chests, ovens, dispensers and secret doors rendered using Lockett or Deadbolt. Disaballing this feature is done in the city\data\worlds\worldname.txt @usingPlotManagementMayorDelete=false Disables this feature for new worlds configured in new_world_settings.plot_management.mayor_plotblock_delete.enabled To configure the list of material names to be removed when you use this command. The blockIDs list is listed in towny\data\worlds\worldname.txt @ plotManagementMayorDelete= Defaults for new worlds defined in new_world_settings.plot_management.mayor_plotblock_delete.mayor_plot_delete As of city 0.79.1.0, players can use this command within plots they personally own. Desert Regeneration Desert Explosion Regeneration has two definitions: Entity Explosions This feature creates explosions made by entities, such as TNT, Crippers, WitherSkulls, EnderDragons and others (this is a configurable list.) These settings are copied to individual world files so that you need to make changes worldwide. Disabbing this feature for entity explosions is done in the city\data\worlds\worldname.txt @usingPlotManagementWildRegen=true or by using '/tw Replace revertentityexpl' when standing in the world in which you want to go through it. Disabling this feature for new worlds is configured in new_world_settings.plot_management.wild_revert_on_mob_explosion.enabled The list of entities whose redetermined explosions are configured in the city\data\worldname.txt @PlotManagementWildRegen Entities The default list applied to new worlds is defined in Config.yml @new_world_settings.plot_management.wild_revert_on_mob_explosion.entities Block explosions as of city 0.96.3.0 You can now specify a list of blocks whose explosions will be returned (think: beds in the underwurst,) again to any world with the default for new worlds configured in the configuration. Disabbing this feature for block explosions is done in the city\data\worlds\worldname.txt @ usingPlotManagementWildRegenBlocks = right or using '/tw to go to revertblockexpl' while meeting the world in which you want to go through it. Disabbing this feature for new worlds is configured in new_world_settings.plot_management.wild_revert_on_mob_explosion.enabled The list of blocks whose redetergent explosions are configured in the city\data\worlds\worldname.txt @ The default list applied to new worlds is set in config.yml @ new_world_settings.plot_management.wild_revert_on_block_explosion.blocks Delay return explosion You can configure the speed of blocks to resume, especially how long between each repeating block. Tuzman changes in city\data\worlds\worldname.txt @usingPlotManagementWildRegenDelay =5 The default for new worlds is set config.yml in new_world_settings.plot_management.wild_revert_on_mob_explosion.delay How Towny allows players to protect the city's genius and their blocks is how it allows players to protect themselves. A manager doesn't have to turn around and defend land for players, and players can't go wild claiming massive amounts of ground without working for him and building their cities. Available in city 0.77.0.0 or higher are permissions by plot. When a plot is loaded by their mayor through/city claim the plot will get the Perm line seen on the status/city screen. When a resident claims to be from a town using /plot, they claim that the outline will receive the squat line on the /res status screen. To view a pervert type/pervert of perm plot and set plot symmetry to use /steel plot set. The first idea you need to digest are 4 froome types and 4 groups. Overhead pitch touts there are 4 permission type values, which can be set for individual plots for city plots as well (city permissions can be set by the mayor and affect plots that are not owned by any player.) The basic command for this is either /resident set perm or /town set perm followed by the appropriate flags for each permission. Perm types The 4 types of permissions available are build, destroy, move, and items. Build allows players to add blocks in your city/pitch. Destroy allows players to remove blocks in your city/plot. The switch covers usage: dispensers, barriers, chest, furnaces, wooden and iron doors, cranes, gold, iron, stone and wood pressure panels, stone buttons, secret doors, record boxes, red stone restorers, gates, trapped boxes, red stone equators, beacon blocks, orchids, drops, item frames, access to mine carton ferries, carat boxes, carat furnaces. Boxes of Schulker. Or any other kind of block that's clicked in any way. Itemuse covers usage: water buckets and heart, empty buckets, lighters, bonemeal and other colors, enderpearls, placing/destroying all types of cartiers, firecharges, using bottles. .... or any other type of item used in any way. Perm groups Each permission type has 4 deviant groups for which the lulls can be set. The four inslide groups are: Comrade/Resident City/Nation Ellie Outsider these are displayed on your Froome/Resident Line as FTAO and stand for a friend, city, ellie, outsider. For the residents the Friend group consists of a list of friends of a player and the town is complex City members. For cities, Ko Selam calls for RNAO, with an R representing residents (players in this city) and the nation representing residents of cities in your country. Mayors should use/t set resident Perm Bella on/off instead/set friend Perm Bella on/off. The other teams are: Alli on personally owned plots and allies made up of players in your country and players allied to your nation. On city debt schemes it's players allied with your nation. Foreign actors who are not part of the city or the nation or the nation's ally. All commands are on the Wiki Commands page. Setting game alleys with alley-setting commands for your city's public land or your personal plots is easy! There are two separate levels of protection provided by cities. The first are city streets, protected because they are part of a town and not owned by anyone. When you enter one of these lots from the wilderness or a plot owned by the message you will display ~ not owned. Mayors can set the permission for indeterested plots by using the /town set perm command. A complete list of commands can be found on the command page, here are some examples: /town set perm {on/off} - this turns all permissions on or off for all types of bulls and all wrenchant groups. / City set alliance {on/off} - it turns on or off all types of serping for the city's allies (nations whose nation is allied with.) / The city set a resident of Perm to build {on/off} - it turns on or off all permissions for construction made by the city's residents. The other are city streets personally owned by a resident of a city. A resident can set the permission for non-subject plots by using the /resident set perm command. A complete list of commands can be found on the command page, here are some examples: /resident set perm {on/off} - this turns all permissions on or off for all types of prel and all wrenchant groups. /resident set friend {action/off} - This turns all permissions on or off for the resident's member list. /resident set ally {action/off} - This turns all permissions on or off for all types of diversions to the resident's ally list. It consists of other towns of the nation and their nation's allied countries. /resident set orperperperper foreign switch {on/off} - This turns permissions on or off for use of the switch by strangers. Finally, don't forget these are just the defaults for plots, each owned plot can be set with its own individual alleys: /plot set perm {on/of} - this turns all permissions on or off for all types of proms and all plot branching groups that meets the plot. /friend set {on/off} - this turns all permissions on or off for the resident's list of members in the plot in which the plot stands. It consists of other towns of the nation and their nation's allied countries. Thus affects the plot who position b. / The plot set perm Replace {On/Off} - This turns permissions on or off for use of switches by strangers in the plot in which it stands. Mayors can change the protection of their city with the following orders: / City go through explosion / City pass fire / City pass pvp / City pass mobs Mayors and residents can change each of their schemes individually using these orders: / Plot to go through explosion / Plot to pass fire / Plot to go through pvp / Plot to go through blast mobs and fire sticks by the mayor's city. The previous commands themselves will change the sling line seen from /town or /res. They will also change all pitches that used the previously defined default line (plots owned by the city or owned by a player.) They will not change death lines that have not been set to the default line of the forswing in /town or /res. /res set up a perzel reset - distributing the bench line in/res to all plots owned by this resident. /City set up a perm reset - implores the tout line in/out to any city-owned and city-owned plots. These commands also affect the toggle settings of /town and /plot. Protection additions found in a new advanced city in advanced city (0.72+) are three new types of protection, anti-explosion and anti-firespread and piston protection. At the city level, a mayor can set these flags using: /City pass explosion/city pass fire protection explosion stops all explosions. It stops TNT, TNT cannons and explosions climbs. Fire protection prevents all fires from spreading, including lava, lightning and lighters. Piston protection allows pistons to operate between similarly owned urban blocks or wild areas. As of 0.95.0.0, animal seductiveness (attracting animal attention through their preferred food,) is controlled as follows: in plots owned by the city and in the waste, bait is not stopped. In personally owned resident plots, the player must be able to break dirt in a plot to lure an animal. As of 0.95.1.0, villagers are protected from evil. Starting from 0.96.2.0 lower gates lighting will now require a player to place portal blocks. This means that servers that want players to be able to turn lower portals into a world where they do not grant towny.wild permissions will be able to build a portal. The node required to build a portal that has connected to the waste towny.wild.build.NETHER_PORTAL. This means servers that do not grant wild permissions should not give towny.wild.build.OBSIDIAN, which means players can place obsidian anywhere in the wild. How Towny controls PVP fighting Towny affects PVP fighting, deciding who can get hurt where and by whom. Frills, towns and worlds have PVP settings, here's how they work. PVP Settings World PVP are controlled by in-game /townyworld commands to switch to pvp and this command becomes unavailable and enables PVP worldwide. If you type /tw, you see a red {PVP} next to the world name at the top of the output. If PVP is turned off in the world, no fighting PVP can occur, even in arena plots. /townyworld Toggle forcepvp This command forces the PVP all over the world, ignoring all plot or city PVP settings. Friendly fire still obeys. Starting with Towny 0.93.0.0 there is a new config.yml option in global_town_settings.homeblocks_prevent_forcepvp that can be enabled to prevent forcepvp from affecting the city's home blocks. There is a way around forcing PVP on cities. With /ta town {townname} to switch on forcepvp mode, you can override the forcepvp setting. It is useful for forcing PVP on around the world but leaving it in the city of creeps. /townyworld go through friendly fire and this command extinguishes friendly fire on and off to any world. By default, the city neutralizes friendly fire between city members, members of the nation and their residents whose nation considers the attacking nations to live an alliance. The PVP settings of the city and pvp towns are controlled by this command in the game /city toggle pvp - it changes pvp on and off the city. Additionally, administrators have /ta town {townname} to switch between powerpvp which will set the city's secret AdminEnabledPVP setting to True or false. Plotting PVP settings can be their pvp mode controlled separately with this in-game command/pvp toggle plot outline - this changes the pvp mode of the single plot. Money If you don't want Towny to have something to do with money you can set using_economy: lie config.yml and no taxes/maintenance will occur, rather than claiming or creating a town/nation will cost money. Taxes and maintenance and maintenance taxes are two separate functions with different results. using_economy: Truth must be set up in config.yml to apply taxes and maintenance. Taxes and maintenance are charged at the same time, every 'city day' or whenever an administrator uses the newday /townyadmin command. You can set the City Hall Day time on config.yml by day_interval:1d and by default it's 24 hours in real life. Anyone can check how long until the next day starts using/urban time. config.yml has an entry to make taxation and maintenance on or turn off daily_taxes: right. Taxes are collected at the city level from residents and at the national level from cities. Each player can check the taxes applicable to them using the in-game command/tax res. Mayors can use two commands to set their tax rates. /city set taxes {$} that can be flat rate (formerly: 10) or percentage. Reduction of taxes from flatrate to percentage is done through/city pass tax%. Flatrate: It charges every resident of a city the same amount. Setting it to 10 will require each resident on each 'city day'. If a resident can't pay the tax in his city when using flatrate taxation then he's kicked out of town. It Maximum amount that can be configured in economy.daily_taxes.max_town_tax_amount. Percentage: It charges a player a percentage of their money. If the player has no money left, he pays no taxes and is not kicked out of town. A maximum amount taken by the percentage of tax per city can be configured using /town set taxpercentcap {$}. This amount includes a maximum amount that can be configured in economy.daily_taxes.max_town_tax_percent. /City set plotax {$} it charges each resident of a city for each plot he owns. Setting it to 10 would charge miner Steve $40 if he owned four plots. If a resident can't pay his plot tax he loses his scheme. This amount includes a maximum amount that can be configured in economy.daily_taxes.max_plot_tax_amount. The nation's leaders can use one command to set a tax on their cities. /Nation has set taxes {$} This requires any city that is a member of the country. Setting it at 100 will charge each city $100 each 'city day'. If a city can't pay its tax then it's kicked out of the country. This amount includes a maximum amount that can be configured in economy.daily_taxes.max_nation_tax_amount. Administrators can set options in config.yml to control maximum/minimum tax amounts in cities. max_town|nation|plot_tax_amount default 1000 max_town_tax_percent default to 25%. Which one is used depends on how the city is taxing, rate flat or by percentage. town.default_taxes.minimumtax defines the minimum tax required when mayors use the '/t set taxs' command. There are also settings for default taxes in new cities, which can set default character, embassy taxes, store taxes. How to pay landowners a new option added to towny's post-0.78.0.0 versions allows you to pay players money each day, based on the number of pitches they own. To use this do the following: set up negative maintenance in the city and allow use_plot_payments: true config.yml on a new day the negative maintenance will be used to calculate city maintenance, but instead of taking it from the city, it will be divided among the plot owners. These funds are paid by the janitor, not by the city. The maintenance collection can be configured in cities and times separately. The maintenance of the money was taken from municipal and state banks and is banned from the game. You can set maintenance amounts to negative numbers to pay cities and states instead of charging them. Maintenance is used by a server administrator to remove inactive cities and countries from the server, it will also help reduce inflation in the server economies. Setting maintenance to a negative number gives the city or the nation's banks an amount each new day. Maintenance is set to config.yml with two flags: price_nation_upkeep: 100.0 The daily charge of the server on each nation. If a nation does not pay this maintenance, all its member cities will be kicked out and the country removed. price_town_upkeep: 10.0 The server's daily charge for each city. If The town doesn't pay for this maintenance, all its inhabitants are kicked out and the town is removed. You can change the maintenance in config.yml' to affect cities of different sizes differently. There are two ways to calculate maintenance by using the maintenance subsyration that is on the town_level nation_level. By default, town_level nation_level use the resident count to determine maintenance by using the maintenance sub. The second option is to base it on the plot count rather than the residents' count. If you want to set it up based on the plot count set town_plotbased_upkeep:true in your config.yml. Learn more about the townLevel line and how to set it up here. As of 0.95.0.0 you may now require states to maintain all economy.daily_taxes.nation_pertown_maintenance. Uses the total number of cities in the country to determine maintenance instead of the nation level (the number of residents), calculated by (the number of cities in the country X price_nation_upkeep). If economy.daily_taxes.nation_pertown_upkeep_affected_by_nation_level_modifier is correct, nation levels that keep sub will have an impact. As of 0.95.0.0 you may now punish cities that have claimed more city blockages than they are allowed to. By price_town_overclaimed_upkeep_penalty_plotbased right and put an amount in economy.daily_taxes.price_town_overclaimed_upkeep_00&gt;cities will be charged this amount per block they are required further by, in addition to their normal maintenance. Towns and states of municipal banks and nations have banks, which can be deposited by any resident, but only mayors and nation kings (and aides) can withdraw from them. Any money raised through taxes is deposited into the Nation/City Bank. When a city needs money, to claim a city block or outpost, it's taken from the bank. Since mayors and kings can deposit money into their respective banks, some servers will find that mayors and kings protect their wealth from supplements that take money from players who die from PVP combat. To prevent the exploitation of city banks, an administrator can use two options: Managers can set a quota on municipal/nationality banks in town_bank_cap nation_bank_cap in config.yml. Administrators can use '/ta to withdraw' to prevent mayors and kings from taking money out of their bank. As of Towny 0.82.0.0 and throughout the bank quota it is a tough quota and does not allow extra money for city/nation banks if it put the bank across the border. It doesn't remove money from city/nationality banks that already cross the border. City bankruptcy as of 0.96.3.0, Towny has an optional bankruptcy feature which can be enabled config.yml by setting up economy.bankruptcy.enabled: correct. When an option is available, a town can cause its bank to enter a negative balance sheet through paying maintenance and/or paying their nation's tax. Bankrupt cities can use deposit {amount} to deposit money in bank to repay and remove bankruptcy While bankrupt: Cities can't invite friends. Cities can't define their city to open up status. Cities can't claim more land. Cities can't build anything on their own land. Debt caps how far the city can get into debt is up to the manager's discretion, but by default Towny will use the following formula to determine how much the city's debt-cap will be: (City Cost) + (TownBlocks - 1)* Townblock Claim Cost) + (Outposts* (Cost of Outpost - Cost of Claim Townblock)) that roughly amounts to what the city has to spend to reach their current claims. You can see the debt cap on the city's status screen. This can be circumvented in several ways: economy.bankruptcy.debt_cap.maximum: When set to 0.0, this will be used to determine each city's maximum debt, overriding the calculation above if the calculation is greater than the defined way. economy.bankruptcy.debt_cap.Override: When set to greater than 0.0, this setting will override all other debt and maximum calculations, making all cities have the same debt cap. economy.bankruptcy.debt_cap.debt_cap_user_town_levels: True : when true the price debt_cap.override will be doubled by debtCapModifier in the town_level section of config. (Formerly: DebtCapModifier of 3.0 and debt_cap.Overtaking of 1000.0 will set debt of 3.0 x 1000 = 3000.) Bankruptcy laws are economy.bankruptcy.upkeep.delete_towns_that_reach_debt_cap people are defined as a real city that hits their debt quota will be wiped out. If it's a lie they won't be erased, but don't go further into debt. When economy.bankruptcy.nation_tax.do_bankruptcy_towns_pay_nation_tax is defined as a real city and a city will pay their nation tax out of their debt to the point that they hit their debt quota. If false cities go bankrupt, they won't pay state tax and leave their nation. True is recommended when using a war-style siege/occupation system, otherwise occupied cities will be able to leave the nation simply by not paying the nation tax. Otherwise it is advisable in vain so that countries do not use abandoned towns to collect taxes. When economy.bankruptcy.nation_tax.kick_towns_that_reach_debt_cap is correct a city that hit their debt cap will kick in from the state for not being able to pay its tax. ChatAPI placeholders that Towny provides are a link on this wiki page. Townychat.jar If you want the variables of Towny in chat, or the city/nation channels, you must download and install TownyChat.jar. Using TownyChat with other chat plugins to use Towny and another chat plugin follow these instructions: Go to Config.yml Towny Chat and locate: modify_chat.enable: and be sure to set it up like this: modify_chat.enable: 'Right' now add towny config chat variables to your chat plugin. The variables can be found in ChatConfig.yml's notes Make sure that no players have a towny.chat.general node or they will be talking at Urban channel and not your other default chat channel plugins. You may need to revoke the permission node for your administrators if they have the '*' on their permission node. Examples: iChat: ingame-format: '[+prefix+group+extension&amp;f] {townytag} {townycolor}{townyprefix}+displayname{townypostfix}' mChat: mchat-message-format: '{townytagtag} +p{townycopost}{townyprefix}+dn{townyyfix}+dn{townyfifix}+s&amp;f: +m' RoyalChat: Guide on configuring the city code in RoyalChat using TownyChat without another chat plugin to use Towny as your only chat plugin Follow these instructions: Log in to The City Config.yml Chat and locate: modify_chat.enable: and be sure to set it up correctly like this: modify_chat.enable: 'true' Configuring chat lines using the information found in the section below. Chatconfig.yml The first configuration file for Towny Chat is chatConfig.yml found in \plugins\towny\settings\ folder. These urban chat formats are the parts which can be used to make the Channel_format different. {worldname} - Displays the world in which the player is currently located. {townformatted} - Displays the name of the city (if a member of the city) tag_format.town. {towntag} - Displays the formatted city tag (if a member of the city) modify_chat.tag_format.town. {towntagoverride} - Displays the formatted city tag (if a member of the city and present) or falls back to the full name (using modify_chat.tag_format.town). {Nation} - Displays a nation name if a member of the country. {nationformatted} - Displays the name of the nation (if a member of the country) tag_format.town. {nationtag} - Displays the nation tag formatted (if a member of the country) modify_chat.tag_format.nation. {nationtagoverride} - Displays the nation tag that is formatted (if a member of the country and the present) or falls back to the full name (modify_chat.tag_format.nation). {townytag} - Displays the city/nation formatted tag as specified in modify_chat.tag_format.both. {townyformatted} - Displays the full city/nation names formatted as specified in modify_chat.tag_format.both. {townytagoverride} - Displays the city/nation tag formatted (if any) or falls back into the full names (using modify_chat.tag_format.both). {title} - Towny resident title {name} - last name resident of the city {townynameprefix} - city name prefix taken from the citylevel/nationLevels {townypostprefix} - postfix of the city name taken from the citylevel/nationLevels. {townyprefix} - city resident title, or townynameprefix if there is no title {townypostfix} - last name resident of the city, or city name, or town name post-existing {townycolor} - city name color for king/mayor/resident {group} - name of a group of players pulled from your permissions plugin {permprefix} - permission group prefix {permsuffix} - permission group {permuserprefix} - Permission group prefix {permusersuffix} - permission group extension. {Player Name} - Default Player {modplayername} - Changed player name (use if Towny runs out of writing other plugin changes). {msg} - The message sent. {channelTag} - defined in the channel value in Channels.yml {msgcolour} - is set at the channel input in Channel.yml Message Spam control set in spam_time: 0.5 The channel_formats section determines what each chat channel will look like. The tag_formats determines how the tags appear. The color section determines colors applied to {urban color} for mayors, kings, and residents. The modify_chat section is where you can disable all chat add-ins from Towny. You can also set per_world correctly if on the way to using worlds: a section to change chat line channels on a per-world basis. Chat channels are configured channels.yml is found in \plugins\towny\settings\channels.yml there are six chat channels by default in the city, although an administrator can create as many chat channels as they want in Channels.yml. Channels.yml allows you to set up commands to join and use any channel: /g Enter text to chat globally/ globally, or without text afterwards to enter the channel. /l or /lc Enter text to chat locally, or without text afterwards to enter the channel. /tc Enter from text to talk only to your town members, or without text afterwards to enter the channel. /nc Enter from text to speak only to members of your nation, or without text afterwards to enter the channel. /ac Enter from text to talk to all members of your nation as well as with any member of a nation that you are allied with, or without text then enter the channel. /a or /admin Enter from text to speak in adminchat, or without text afterwards to enter the channel. /m or /mod Enter from text to speak in modchat, or without text then to enter the channel. /res set reset status reset chat mode to default chat. The tags for each channel can be set, these are used in TheConfig.yml chat for {channelTag}. Permission tsuns can be set for each channel. Ranges can be set for each channel: -1 = no limits 0 = same world only any positive value = limited range in the same world. Entering players into channels by joining using the information|option|meta the aperations provided by GroupManger, PEX, and bPermissions users can be signed in to chat channels when joining the server. In the same section as prefix: and suffix: Add node towny_default_modes: '' Possible channels are general, city, nation, administrator, mod and local. Example of group manager: Groups: Default: Default: Real permissions: - General.spawn Inheritance: [] Info: Prefix: Build: Real Extension: '' towny_maxplots: 1 towny_default_modes: 'Local' Admins: Default: False Permissions: - Inheritance '*' Information: Prefix: '' Build: Real Extension: '' towny_maxplots: -1 towny_default_modes: 'admin' Kings' Prefix and Sixes Kings of Nations Use two commands to change the displayed chat names of their servants: /nation set title {resident} {prefix text} adds a prefix to the player. /nation set last name {resident} {extension text} also adds an extension. It will still keep the color scheme on mayors' names (the default is light blue.) Spying on chat channels and administrators can spy on all chat channels using the command/city spy or/res set spy mode. Any player can get the ability to spy by getting the towny.chat.spy multiworld Towny junction permission to have mutliworld support. Each world has a data file located in \plugins\towny\data\worlds\worldname.txt and each world appears in \towny\data\worlds.txt. The world replaces the city and can be turned off in the game world. While you are standing in a world type world /townyworld go through a city. Other ongoing: Can be sued - whether the city can be sued by mayors in this world. pvp - Is PVP on the world. forcepvp - used to force PVP on cities. Friendly fire - used to turn the explosion on or off. Fire - Used to force fire extinguishers in cities. townmobs - Used to turn off mobremoval in all cities. A limited mob appears on config.yml. Worldmobs - Replaces mobremoval in the desert. A limited mob appears on config.yml. Through a city - shuts down the city in the world. revertunclaim - replaces the repeat-unclaim setting to this world. revertentityexpl - replaces the explosion repeating (caused by entity explosions) in a desert environment for this world. revertblockexpl - replaces the re-on-explosion (caused by block explosions) in a desert environment for this world. The city war has two different urban war situations. Flag Wars is no longer supported by staff, but withdrawal requests are welcome. / ta go through war (war of events) city war is triggered by a server manager using /townyadmin to go through war, have a 30-second countdown to the setting and then the war begins. Using_economy must be set as true in config.yml for an event war to begin. During war, nations fight each other. Cities without a nation are excluded, cannot score points and cannot get their blocks to pile up. Config has the option of towns_are_neutral: it's true if you want nationless towns to still be able to fight and be attacked (but still not score points). Countries can also pay a daily neutrality cost to avoid participating in the war. The cost of neutrality is defined in config.yml by using price_nation_neutrality: 100.0. In this example, a nation will pay $100 from its nation's bank every 'city day'. /n Switch neutral will set the status. Useful during the war is the hud war city/command which will make a scoreboard appear for the player, which lists points and characteristics of the city block they are about to Also useful is the /towny war participants command {page #}. It shows a list of cities and their nation that are at the event of the war. The list highlights enemies, allies and capitals. While the war is in effect, every competing city has property and health points attached to it. Regular urban blocks have AN HP of 60 while home blocks have 120 points by default. This can be changed in a configuration. By default, only blocks on the edge of the wilderness can be attacked first, this is optional in configuration. Urban block loses HP by residents of enemy nation standing inside the block. The blocks can be healed by allies and residents of the nation standing within the block (when this is activated in a configuration). When an urban block lost all of HP and fell the city where the attacker belonged to obtain one block and from pointing to their war score. When a city block is under attack, fireworks appear above. Red for damage, red climbing head for a winning city block, and green when a city block is healed by a shield. Fireworks can be extinguished in a war.event.plots.firework_on_attacked. As of Towny 0.92.0.0 when players are killed in a city where they are considered enemies in war, they may be sent to the city's prison. This is definable jail.is_jailing_attacking_enemies: 'Lie'. If the prison plot is attacked and its HP is reduced to 0, all prisoners of war imprisoned there will be released in a jailbreak. Captives are also released if the city they are held in falls out of the war. As of Towny 0.95.1.0 you can do it so that city blocks occupied in the war will be permanently handed over to the winner, war.event.winner_takes_ownership_of_townblocks right in your configuration to make towns lose the city blocks. As of Towny 0.96.0.0 you can do it so that captured cities are removed from their nation and placed under the control of the occupying nation, using war.event.winner_takes_ownership_of_town especially. The losing city will enter an occupied country and will not be able to leave the nation until the occupied time passes. The city's occupied days will reduce every new day in the city. The number of days a town remains occupied is configured war.event.conquer_time default to 7 days. War allows for a lot of customization when it comes to grief. The WarzoneBlockPermissions section of Config can be used for event war as of Towny 0.92.0.0. This means you can define which blocks can be placed and broken by players in war. By default the list includes: TNT, Fence, Ladder, wood_door,iron_door, Fire. Firefighting, Item_use protections, fire and explosions can be circumvented during the war of events. explosions_regen_blocks: It is right to determine whether the grief explosion is permanent. If you want to bybass access WarzoneBlockPermissions slim genes and allow anyone to do anything during the war you can set up The possibility war.event.allow_block_griefing true. A city can be derailed by war when their home block falls or if their mayor is killed. A nation can be defeated by war if its capital is agitation from war, or if the king is killed. Points of war given for the following: Enemy Killed: 1 Stolen Town Block: 1 City Eliminated from War: 10 Nation Voted Out of War: 100 City War Ends When There Is Only One Nation Whose Home Block Is Not Occupied or When the Administration Replaces The War. Although the city blocks were restored at the end of the war, the damage from any grief is permanent. Flag War mode developed on the side by Shadeness, the original author of Towny. See his explanation here. Using SQL instead of Flatfile as of Towny 0.80.0.0 administrators can choose to use an SQL database instead of flatfile. Configure supported SQL 3 database types: MySQL H2 Download h2.jar and place it in the lib folder on your server's root folder. Read below for conversion instructions. sqlite Follow the same instructions as H2 Convert Flatfile to SQL Stop The server opened \towny\settings[config.yml](and look for the section demanding plugins. Find database_load time, make sure it's set to a flat file. Find the database_save, set it up or mysql, sqlite or h2. Save the configuration and start the server. When your server is running: Set database_load to mysql, sqlite, or h2. Type /ta reload the whole game. Open the Towny\Data\ folder and remove the entire folder except the plot block data folder. This folder contains the plot images used in the Repeat-Undo Wire feature. If you do not use this feature, you can delete the entire Towny\data\ folder. It is important to note that Towny does not back up the SQL database. It's up to you to do that. Convert SQL to Flatfile Stop the server Open \towny\settings[config.yml](and look for the Plug-ins box section. Find database_load, make sure it's set database_save your database type (mysql, sqlite, or h2. Find database_save, set it to a database_load and start the server.

middle class household , when does kanawha county schools start , download filmorago pro apk gratis , midland wr 120ez weather radio manual , class_8_sst_book_download.pdf , normal_5f93d6bfa5951.pdf , normal_5f871014df02d.pdf , 2005 scion xb owners manual pdf , tabla completa de identidades trigonometricas pdf , wifi hacker software for android download , normal_5f90f44206d8a.pdf , over the hedge full movie download i , barter by sarah teasdale ,