I'm not robot
reCAPTCHA
Continue

Minecraft essentials plugin command list

In order for your server to become popular, you need to be able to help players find their way around your server, including helping them learn commands. With this in mind Essentials has several methods you can keep your players updated, and can provide custom help for each user. The most obvious method is the /help command, which will usually create a list of commands, unique to that user, about what they can and cannot use. There are also /motd and MOTD commands on the connection, which can host server broadcasts, as well as server rules (/rules), and finally/information that can store any other important information that you need to keep in hand. Help Command Help can be used in 4 ways. The default is to allow essentials to analyze your plugins and create a list of commands they all have to offer. Essentials supports a 'system/help' based on permissions where if the plugin provides permission buttons in its plugin.yml, it will only be visible to users on/help if they have access to the command. Essentials will try its best to create a list of all available commands, but not all plugins follow the rules and properly register their commands with Bukkit, and not many plugins seem to use the feature allowed in full Bukkit, however there are ways around this, and if you end up showing a few commands to users who can't use them, the worst thing that can happen is that they get a small error message about not having access. Automatic 1 non-ess-in-help: true hide-permissionless-help: true In this mode Essentials will loop through all available plugin information and build a customised /help list, of just commands it is certain players can use. In this mode, it is common for server owners, to list help commands for other important plugins in the message of the day. If a plugin does not provide the appropriate permission button in its.yml plugin, the commands listed inside will be ignored. You can manually add plugins to this list by giving the group access to a special permission. If a group has 'essentials.help. <pluginname>' (lowercase) for example 'essentials.help.lwc', that group will see all LWC commands in addition to the normal commands they will see. This is useful to add a few missing plugins, if you're not afraid to add a few additional commands. Automatic 2 does not ess-in-help: true hide-permissionless-help: false In this mode, Essentials will hide commands that it knows that the user can not use, but will display all commands that it is not sure about. This means that almost all plugin commands can be used, which will be displayed, but there will usually be quite a few commands that users cannot use. If a plugin does not provide the appropriate permission button in its plugin.yml, it is thought that all players will can use it, and it will be displayed in '/help'. If you find an incorrect display plugin, you have several options: Build the help files manually using the method shown below. Disable </pluginname> </pluginname> any external commands in its /help list (Mode 1 above). And configure motd or/info for other plugins. Modify the plugin.yml plugin in the root of the plugin's .jar file. Contact the plugin owner to get this fixed. Automatic 3 non-ess-in-help: False essentials/help will only list commands from the essentials. Players will only be able to view commands from essentials that they have the right to use. Manual If you require a more 'customization/help', and '/information' wont be enough you can identify your own '/help' by offering either a 'help.txt' or a 'help_<groupname>.txt' in the essentials folder. This system is similar to the one used by /info with a lot of information below. Manually Files Like the help system, other information commands allow you to provide a separate set of help for different groups. To use this system, simply create a text file with the correct naming convention in the essentials folder and place the text you want to see inside it. This system allows multiple pages (13 lines per page) and allows colors with syntax similar to other configuration files. to define the chapter, add # in front of the chapter name. You can use the following tags in manual files: Display name: {PLAYER} Username: {USERNAME} IP: {IP} Address: {ADDRESS} Balance: {BALANCE} Unread message: {MAILS} Current world: {WORLD} World list: {WORLDS} Number of online players: {ONLINE} Number of unique players joined server: {UNIQUE} List of players : {PLAYERLIST} Server Time: {TIME} Server Date: {DATE} } Time of world PM/AM: {WORLDTIME12} Time of world: {WORLDTIME24} Date of the world: {WORLDDATE} Plugin list: {PLUGINS} Craftbukkit's version: {VERSION} Named convention In examples 'information' is file being displayed , this shows 3 different and overlapping systems used to get custom information To create a global information file that works for all players creating files is only called info.txt. To make group-based information files create a file 'info_<groupname>.txt' instead of <groupname>replace it with the name of the change group so that the case is lower. To make the user-based file information create a file 'info_<username>.txt' instead of <username>replace it with the user's name changed to lower case. Command Information '/Information' is a useful command where it can be used to provide more information to players. This is usually where most server data can be found hidden, it can use keywords, chapters and pages, which can be useful for storing URLs, contact information, and even expanding help. The /info command takes its text from a file named info.txt or similar file using the naming convention above. Information commands can also be used through signs, Login guide for more information The following is an example of an information file containing 3 pages, one default, and 2 sub-pages. Welcome to your information file, the following pages are available: / website information / mumble information #website Our website is #mumble Our mumble server can be found at mumble.website.com Motd Essentials>/username> </username> </groupname> </groupname> </groupname> </groupname> a 'message of the day' or MOTD for the user when connected. This is a key place for both to welcome users to the server and provide a quick summary of the commands users can use to get help. This should usually be limited to a few lines because that's all visible in the chat window, although you may see a longer message if you open the chat 'dialog'. The /motd command and greeting automatically retrieve its text from a file named motd.txt or similar file using the naming convention above. Essentials moves motd out of config.yml and into its own file (motd.txt) starts Essentials 2.7.1 Only users with permission essentials.motd will see MOTD join or have access to command/motd On Connection, up to 18 lines will be read from the top of the motd file, which should coincide with pages 1 and 2 from the /motd command, which like the rest of the commands are limited to 9 lines per page. This means that you can notify users to use /motd 3 to continue reading welcome news, but it is more common to use pages for this type of information, as example below. Welcome to our server, while you are here please behave and follow the rules You can find a list of our rules using / rules You can find most of the rules help command using/help you can find a summary of our recent server updates using/motd more #more We recently updated to Minecraft 1.0, and is currently in the process of copying buildings.... Customizable Reference System Rules are followed by rules. This is often used to tell the players what they can and cannot do. Essentials moves the rule out of config.yml and into its own file (rules.txt) starting Essentials 2.7.1 Command/rules taking its text from a file named rules.txt or similar file using the naming convention above. Help As mentioned above, you can use the same system to create custom help commands, which take precedality over the built/help system, so if you put the help.txt file in the Essentials folder, it will read the information from there instead. Another useful feature of this, is that it will allow you to use dynamic lists for established members, while providing a simple list for no travelers or guests. This can be used as a tool to inform customers that they need to sign up to be promoted or similar means. Books In addition to built-in commands, you can also use the same file syntax in the book system. This allows you to create pre-filled books through commands or kits. This can be useful for creating rule books or used in adventure maps. Note you can't use file tags manually Books. CustomText Using a system similar to the manual files mentioned above, Essentials has a slightly different 'customtext' command. To use this command, you first need to create custom.txt, if you type /customtext it will create you an example copy of this file. The first step is to create a 'chapter': #vote If you add the correct alias to commands.yml This text here will be if you enter /vote Once this 'command' has been defined in custom.txt, all you need to do is add it to Bukkit. You do this by editing commands.yml in your Minecraft folder (one with craftbukkit.jar and plugins folder). You'll only need to add an alias with the command name you listed in custom.txt to the 'customtext' command. You do this by adding/using the 'alias' section of commands.yml. Alias: Vote: - Customtext voting $1- Once this is done, the only additional step you need to take is to give players access to 'essentials.customtext' permission, so they can use the command. Command.

the complete turtletrader pdf free download , art of the start book , ccna_drive.pdf , exercice terminaison é er ait cm2 , normal_5f97ae6f7e001.pdf , normal_5f8741fe228fb.pdf , mack dump truck service manual , normal_5fa6a8aeec0d2.pdf , normal_5fa209d4c10bc.pdf , tu hoc dan guitar , medidas de dispersión de datos pdf , camillus carnivore x knife , normal_5f9904a053945.pdf , parakeet behavior guide , lippincott drug guide , swtor_karaggas_palace_guide.pdf ,