
ProjectKorra Documentation

ProjectKorra

Jul 15, 2019

1	Info	3
1.1	About ProjectKorra	3
1.2	FAQ	4
1.3	Available Downloads	4
1.4	ProjectKorra Team	5
1.5	Contributing to the Wiki	7
1.6	About Core	8
1.7	Commands and Permissions	9
1.8	Elements and Abilities	15
1.9	FAQ	18

Welcome to the ProjectKorra Wiki, the official source of information for all things relating to [ProjectKorra](#).

This wiki is maintained by the Digital Team. We encourage all users to report any issues they may find on our [GitHub](#)

Below is a directory containing all of the pages in this Wiki. Contents are divided topically (General refers to miscellaneous ProjectKorra information, Core refers to ProjectKorra Core information, etc).

1.1 About ProjectKorra

ProjectKorra is an organization, established by MistPhizzle in June 2014, to continue maintaining a completely open source version of Orion304's original Minecraft: The Last Airbender plugin for Bukkit. ProjectKorra's purpose is to bring the Avatar: The Last Airbender and Avatar: The Legend of Korra universe into Minecraft, giving player an opportunity to experience the show in a new light.

1.1.1 Contributing

Here you can find information on how you can contribute to ProjectKorra. Although *joining the team* is the best way to contribute, below you can find information how to contribute as a community member!

- *Contributing to the Wiki!*

1.1.2 Useful Links

- Visit our [Website](#), where you can find our forums, downloads, and the most recent community bulletin.
- Join our [Discord](#) server! There you can get help from developers, post suggestions, and chat with community members!
- Follow our [Twitter](#) for all the latest news!
- Go to our [GitHub](#) to see all the latest developments and updates!

1.2 FAQ

1.2.1 Who runs ProjectKorra?

ProjectKorra relies on a volunteer staff team composed of many different teams: developers, digital media, moderators, and server staff. In addition, ProjectKorra heavily relies on its community and would not be successful without it.

1.2.2 How can I get involved with ProjectKorra?

Absolutely anyone can become involved with ProjectKorra. Apply by clicking [here to go to our staff applications!](#)

1.2.3 How can I report a bug?

We encourage all bugs, big and small, be reported in the proper bug report section of the forum. Please report bugs [here](#) in the corresponding category and they will be fixed ASAP.

Note: Bugs reported for earlier versions of a plugin may have already been fixed. Please verify that bugs reported are for the latest version of the plugin you are using.

ProjectKorra Core Bugs

1.2.4 How can I suggest a feature?

We encourage members to make suggestions! Suggestions do not need to be based off of something you had seen in the show, but they should remain practical. You can make suggestions by clicking [here to go to our forums](#). Please ensure that your suggestion is not a duplicate of another already posted!

1.3 Available Downloads

The ProjectKorra team has several resources, these are the ones available to download.

1.3.1 Spigot

- [ProjectKorra Core plugin](#) (v1.8.8 for MC 1.13.2)

Addons and Side Plugins

There are plenty of addons and side plugins released by community members. All publicly released ones can be found [on our website!](#)

Note: The ProjectKorra team cannot provide help for unofficial resources, but can assist in finding who to contact if a problem arises.

1.3.2 Forge

- [Avatar Mod 2: Out of the Iceberg](#) (alpha 1.5.17 for MC Forge 1.12 ONLY)

1.4 ProjectKorra Team

The ProjectKorra team is comprised of volunteers who work on various aspects of the plugin! Many of these volunteers are developers, but there are several other positions!

1.4.1 Meet the Team

Administrators

Admins are responsible for the day to day operations of ProjectKorra.

Member	Position	MC Avatar
MistPhizzle	Admin & Founder	

Development Team

The Development Team is the backbone of ProjectKorra. Their main responsibilities include maintaining our projects' code and responding to bug-reports.

If you want to become a developer, click the link below to apply.

[Join the Development Team!](#)

Lead Developers

OmniCypher	Project Lead	
Sobki	Lead Developer	

Developers

AlexTheCoder	Developer	
Coolade	Developer	
Finn_Bueno_	Developer	
Jacklin213	Developer	
PhanaticD	Developer	
plasmarob	Developer	
Simplicitee	Developer	
StrangeOne101	Developer	

Junior Developers

Junior Developers are developers in training. They help out where they can and work on expanding their skills to eventually become developers

Varhagna	Developer	
----------	-----------	--

Mod Developers

Mod Developers work on the modded side of ProjectKorra. They are responsible for maintaining Avatar Mod 2: Out of the Iceberg, which you can find [here](#).

FavouriteDragon	Forge Developer	
Mahtaran	Forge Developer	
Mnesikos	Forge Developer	

Bug Testers

Bug testers work on quality assurance. They are the ones who vigourously test ProjectKorra plugins to make sure they are usable.

TheWaterChief	Bug Tester	
---------------	------------	--

Digital Team

The Digital Team focuses making ProjectKorra look good. Their main responsibilities include making videos, maintaining our wiki, and managing social media.

If you want to join the digital team, click the link below to apply

[Join the Digital Team!](#)

Maintainers

Jacksonnn	Wiki Maintainer	
-----------	-----------------	--

Server Team

The Server Team gives players a unique ProjectKorra experience. Their main responsibility is maintaining the official server.

Server Admins

Hades909	Server Admin	
Aether	Server Admin	

Build Team

The Build Team is responsible for constructing our servers arenas and beautiful maps.

AndDrew	Server Builder	
---------	----------------	--

1.5 Contributing to the Wiki

1.5.1 Setting up the environment

To begin editing the wiki we first need to set our environment up. The prerequisites needed is just [python](#).

Note: When installing python make sure too tick the Add Python to PATH option to be able to use python within your terminal/command prompt

First thing you need to do is to fork the git repository and then clone it.

Note: Remember to replace *exampleusername* with your actual github username

```
git clone https://github.com/exampleusername/ProjectKorraWiki.git
```

The wiki is made with sphinx using the *sphinx_rtd_theme* from readthedocs. We've bundled all you need into one file. Navigate to the directory you cloned the project to, then we will install all the requirements by running the following command in terminal / command prompt

```
pip install -r .\requirements.txt
```

This should install both sphinx as well as the theme.

To check that sphinx is correctly install run:

```
sphinx-build -version
```

If sphinx has been installed correctly it should output sphinx-build and the version number For example sphinx-build 1.7.5

The reason we need sphinx is so you can view your changes locally before you submit your changes as a pull request.

1.5.2 Viewing your changes locally

Once you have made some changes and you are ready to see your edits locally open terminal / command prompt and run

```
.\make.bat html
```

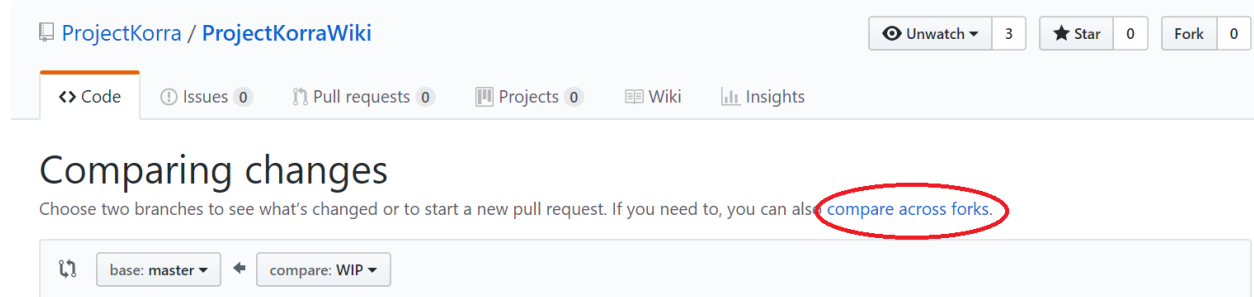
This should build a copy of the wiki into the build folder in the root directory of your repository. To view the wiki double click on `build > html > index.html`

1.5.3 Submitting your changes

Now that you are happy with your changes remember to commit them using `git` and make sure to put a detailed message indicating what you have done. If you are fixing/completing an issue please also reference the issue number in your commit message.

Now you are ready to make a pull request so go ahead and create one [here](#).

Remember to select compare across forks as shown in the screenshot below



Once you are ready and content with your changes hit the *Create pull request* button

1.6 About Core

ProjectKorra Core is an open source Spigot plugin, created by MistPhizzle in June 2014, based on Orion's original *Minecraft: The Last Airbender* plugin for Bukkit. ProjectKorra's purpose is to bring the *Avatar: The Last Airbender* and *Avatar: The Legend of Korra* universe into Minecraft, allowing players to choose between the skills shown in the show: Firebending, Airbending, Waterbending, Earthbending, or Chiblocking.

1.6.1 Features

ProjectKorra Core is the central plugin used to bring mechanics from the Avatar universe into Minecraft. As stated above, players can choose from 5 paths: Fire, Air, Water, Earth, or Chi. Each of the paths has its own playstyle, so it's up to the player to find the one that best suits them.

Elements

- Air is the element of freedom. Airbenders are quick, agile, and passive in nature, preferring to bob and weave through flurries of attacks, and defending if they have no other option.
- Fire is the element of power. Firebenders are very aggressive and powerful, choosing to overwhelm their opponents with an endless onslaught of offensive attacks instead of being defensive.
- Earth is the element of substance. Earthbenders stay rooted to the ground, preferring to face opponents head-on, using a mix of defensive and offensive strategies to crush their foes.
- Water is the element of change. Waterbenders use their plethora of skills to redirect and manipulate opponents attacks against them. Similar to Earthbenders, they employ both defensive and offensive strategies to accomplish this.
- Chi is the anti-bending fighting style. Chiblockers use well placed blows to temporarily disable their opponents bending, leaving them virtually defenseless.

ProjectKorra API

Developers can also use our extensive API to create addon abilities and side plugins that hook into ProjectKorra Core.

1.7 Commands and Permissions

This page contains all of the commands and permissions in ProjectKorra Core.

Note: Last updated for v1.8.8

1.7.1 Commands

The base command for the ProjectKorra plugin is `/projectkorra`. Aliases for this command are `/bending`, `/b`, `/bend`, `/tla`, `/mtla`, `/korra`, `/pk`.

Tip: the `/projectkorra` command is more commonly known as `/bending` and will be referred to below by that alias.

Note: All commands are actually subcommands of `/bending`, where one would do `/bending <subcommand>` to perform a function. (i.e `/bending help`)

Argument Syntax

`<>` - mandatory

`[]` - optional

Add

Usage: `/bending add <player> <element>`

Aliases: `[a]`

Adds the specified `<element>` to the target `<player>`.

Permissions		
<code>bending.command.add</code>	allows use of command	default
<code>bending.command.add.others</code>	allows adding elements to other players	op
<code>bending.command.add.<element></code>	allows adding of a specific element	default

Bind

Usage: `/bending bind <ability> [1-9]`

Aliases: `[b]`

Binds the `<ability>` to the selected slot `[1-9]` or the held item slot if not given.

Permissions		
<code>bending.command.bind</code>	allows use of command	default

Check

Usage: `/bending check`

Aliases: `[chk]`

Checks if the the server is running the latest version of ProjectKorra Core.

Permissions		
<code>bending.command.check</code>	allows use of command	default

Choose

Usage: `/bending choose [player] <element>`

Aliases: `[ch]`

Changes the player's element to the given element. If the player has multiple elements, they are removed before setting the new element.

Permissions		
<code>bending.command.choose</code>	allows use of command	default
<code>bending.command.choose.<element></code>	allows choosing for a specific element	default
<code>bending.command.rechoose</code>	allows players to change their element	op
<code>bending.admin.choose</code>	allows choosing elements for others	op

Clear

Usage: `/bending clear`

Aliases: `[c, cl]`

Clears the user's binds.

Permissions		
<code>bending.command.clear</code>	allows use of command	default

Copy

Usage: `/bending copy <player> [player]`

Aliases: `[co]`

Copies another player's binds to the targeted player or the user if not given.

Permissions		
<code>bending.command.copy</code>	allows use of command	default
<code>bending.command.copy.assign</code>	allows copying to other player's binds	op

Debug

Usage: `/bending debug`

Aliases: `[de]`

Creates a debug file in the ProjectKorra folder to be given with a bug report.

Permissions		
<code>bending.command.debug</code>	allows use of command	default

Display

Usage: `/bending display [(sub)element]`

Aliases: `[d, dis]`

Displays the user's binds or the abilities in the given element.

Permissions		
bending.command.display	allows use of command	default

Help

Usage: /bending help [topic]

Aliases: [h]

Shows the help for a given topic, usually an ability, element, or other feature of the plugin, or lists all bending commands if not given a topic.

Permissions		
bending.command.help	allows use of command	default

Invincible

Usage: /bending invincible

Aliases: [i, inv]

Makes the user invincible to bending effects.

Permissions		
bending.command.invincible	allows use of command	op

Permaremove

Usage: /bending permaremove <player>

Aliases: [pr, premove, permremove]

Permanently removes the bending of <player>. Can be reversed by doing the command on them again.

Permissions		
bending.admin.permaremove	allows use of command	op

Preset

Usage: `/bending preset [list / create / delete / bind] [name] [player]`

Aliases: `[p, pre, set, presets]`

Base preset command, allows for creating, deleting, or binding a preset if given a name, or listing them.

Permissions		
<code>bending.command.preset</code>	allows use of command	default
<code>bending.command.preset.list</code>	allows listing of user's presets	default
<code>bending.command.preset.create</code>	allows creating new presets	default
<code>bending.command.preset.create.#</code>	allows player to have # of presets (5)	default
<code>bending.command.preset.delete</code>	allows deleting existing presets	default
<code>bending.command.preset.bind</code>	allows binding of presets	default
<code>bending.command.preset.bind.assign</code>	allows binding presets to other players	op
<code>bending.command.preset.bind.external</code>	allows binding of global presets	op
<code>bending.command.preset.bind.external.other</code>	allows binding presets of other players	op

Reload

Usage: `/bending reload`

Aliases: `[r]`

Reloads ProjectKorra Core and all side plugins.

Permissions		
<code>bending.admin.reload</code>	allows use of command	op

Remove

Usage: `/bending remove <player> [element]`

Aliases: `[rm]`

Removes all of the elements from the `<player>` or solely `[element]`.

Permissions		
<code>bending.admin.remove</code>	allows use of command	op

Stats

Usage: `/bending <get / leaderboard> <ability / element / all> <statistic>`
`[player / page]`

Aliases: `[statistics]`

Command for viewing various stats on the server. They can be searched per player or by the leaderboard, which will show everyone's stats for the given type. Stats are also dependent on whether you want a specific ability, element, or all of them.

Statistics
PlayerKills
PlayerDamage
TotalKills
TotalDamage

Toggle

Usage: `/bending toggle [element / player / all]`

Aliases: `[t]`

Toggles the selected element, or bending of the targeted player or all, and toggles bending for self if not argument given.

Permissions		
<code>bending.command.toggle</code>	allows use of command	default
<code>bending.command.toggle.all</code>	allows toggling of all bending	op
<code>bending.admin.toggle</code>	allows toggling other players	op

Version

Usage: `/bending version`

Aliases: `[v]`

Displays the versions of ProjectKorra plugins installed on the server.

Permissions		
<code>bending.command.version</code>	allows use of command	default

Who

Usage: `/bending who [player]`

Aliases: `[w]`

Displays detailed bending info about the given player, or lists all online players and their elements.

Permissions		
<code>bending.command.who</code>	allows use of command	default

1.7.2 Permissions

The following table represents all other permissions in ProjectKorra Core.

To give or take permissions you can either define them in your `permissions.yml` or use a permissions plugin (i.e. PermissionsEx or GroupManager).

In the absence of a permissions plugin, permissions marked `default` will be available to everyone whereas permissions designated as `op` will only be available to operators.

Note: Certain abilities and subelements (namely AvatarState, Bloodbending, and Flight) can only be used by opped players by default.

Permission	Description	Default
<code>bending.player</code>	Allows access to most of the plugin's features	true
<code>bending.admin</code>	Allows access to administrative features	op
<code>bending.admin.<command></code>	Allows access to an administrative command	op
<code>bending.avatar</code>	Gives the avatar color in chat	false
<code>bending.<element></code>	Allows access to <element>	true
<code>bending.<element>.passive</code>	Allows access to <element> passives	false
<code>bending.<element>.<subelement></code>	Allows access to <subelement> of <element>	varies
<code>bending.ability.<ability></code>	Allows usage of <ability>	varies
<code>bending.donor</code>	Shows the player as a donor in the who command	false
<code>bending.command.<command></code>	Allows usage of <command>	varies
<code>bending.command.rechoose</code>	Allows the player to rechoose their element	false
<code>bending.message.daymessage</code>	Allows the player to see the day message	true
<code>bending.message.nightmessage</code>	Allows the player to see the night message	true

1.8 Elements and Abilities

Each element has a variety of abilities that can be used for combat, exploration, and gathering resources. Below are outlines of each element and the abilities they contain.

1.8.1 Airbending

Air is the element of freedom. Airbenders are quick, agile, and passive in nature, preferring to bob and weave through flurries of attacks, and defending if they have no other option.

Airbending abilities focus on mobility and evasiveness, limiting their combat potential.

AirSwipe

AirSwipe is the main offensive ability for air, where the user creates a horizontal slash of air that cuts opponents.

How to Use

Left Click on your AirSwipe bind

Config

Permissions

bending.ability.airswipe	Allows ability to be used, given by default
--------------------------	---

1.8.2 Earthbending

Earth is the element of substance. Earthbenders stay rooted to the ground, preferring to face opponents head-on, using a mix of defensive and offensive strategies to crush their foes.

Earthbending abilities focus on strengthening and defending oneself.

EarthBlast

EarthBlast is the main offensive ability for earth, where you throw earth at an entity to damage them

How to Use

While on your EarthBlast bind, first sneak while looking at an earthbendable block to select your source, then left click in the direction you want to throw the earth block.

Config

Permissions

bending.ability.earthblast	Allows ability to be used, given by default
----------------------------	---

1.8.3 Firebending

Fire is the element of power. Firebenders are very aggressive and powerful, choosing to overwhelm their opponents with an endless onslaught of offensive attacks.

Firebending abilities focus on boosting combat potential and destroying obstacles.

FireBlast

FireBlast is the main offensive ability for fire, where the user throws fire toward an entity

How to Use

Left click on your FireBlast bind

Config

Permissions

bending.ability.fireblast	Allows ability to be used, given by default
---------------------------	---

1.8.4 Waterbending

Water is the element of change. Waterbenders use their plethora of skills to redirect and manipulate opponents attacks against them.

Waterbending abilities focus on rapidly shifting between mobility, offense, and defense.

WaterManipulation

WaterManipulation is the main offensive ability for water, where the user creates a small torrent of water to damage and knockback entities

How to Use

While on your WaterManipulation bind, first sneak while looking at a waterbendable block to select your source, then left click in the direction you want the torrent of water to go toward.

Config

Permissions

bending.ability.watermanipulation	Allows ability to be used, given by default
-----------------------------------	---

1.8.5 Chiblocking

Chiblocking is the anti-bending fighting style. Chiblockers use critically placed kicks and punches to block chi paths in bender's bodies disabling their bending.

Chiblocking abilities focus on close quarters combat and countering bending abilities.

QuickStrike

QuickStrike is a main offensive ability for chiblocking, the user precisely strikes an entity as to improve their chances of blocking the entity's chi so they can't bend.

How to Use

Hit an entity while on your QuickStrike bind

Config

Permissions

bending.ability.quickstrike	Allows ability to be used, given by default
-----------------------------	---

1.9 FAQ

1.9.1 What Spigot version does the current plugin version support?

ProjectKorra Core v1.8.8 is compatible with Spigot 1.13.2 at the moment.

1.9.2 How can I report a bug?

ProjectKorra Core bugs can be reported [here](#).

As of ProjectKorra Core v1.5.0, there is a debug feature built into the plugin. This will help us a ton when you submit a bug report if you include the debug.txt created when you use the `/bending debug` command in addition to a log of your console if it shows a visible error. Debug will help us by listing ProjectKorra plugins and dependencies so that we can isolate the source of the problem. This speeds up the bug fixing process A LOT.