

GetPlayerVersion

Note: You can also compare if a version is higher than another using the < or > operators.

This function gets the client version of the specified player as a sortable string. The string is always 15 characters long and is formatted as follows:

- 1 character representing the major version
- 1 dot character
- 1 character representing the minor version
- 1 dot character
- 1 character representing the maintenance version
- 1 dash character
- 1 character representing the build type
- 1 dot character
- 5 characters representing the build number
- 1 dot character
- 1 character representing the build revision

An example of a version string would be: 1.0.4-9.01746.0

Where the first three numbers represent the major/minor/maintenance version, i.e. 1.0.4
The fourth number is 9, which means it's a release build, (Development and beta builds have lower numbers here)
And the fifth and sixth numbers represent the build number.

Syntax

```
string getPlayerVersion ( player thePlayer )
```

OOP Syntax [Help! I don't understand this!](#)

Method: *player.getVersion(...)*

Variable: *.version*

Required Arguments

- **thePlayer:** The player whose client version you wish to get.

Returns

Returns a string containing the client version, or false if the player is invalid.