ctrl K

Overview

Functions

Importing The Module

Spell Availability 🔲

Spell Cooldown 🔀

source, destination)

Line of Sight Checks 👁

id, caster, cast\_position)

get\_spell\_cost(spell\_id)

Casting Readiness 🗸

Resource and Cost Checks 💍

can\_afford\_spell(unit, spell\_id,

is\_spell\_castable(spell\_id, caster,

How To - Check If You Can Cast A Spell

target, skip\_facing, skips\_range)

caster, target)

spell\_costs)

Examples

has\_spell\_equipped(spell\_id)

is\_spell\_on\_cooldown(spell\_id)

is\_spell\_in\_range(spell\_id, target,

is\_spell\_within\_angle(spell\_id, caster,

target, caster\_position, target\_position)

is\_spell\_in\_line\_of\_sight\_position(spell\_

is\_spell\_in\_line\_of\_sight(spell\_id,

Range and Angle Checks 🎯

How To - Check If You Can Cast A Spell 🎯

This is the recommended way to check if you can cast a spell. Just check the last two parameters (skip\_facing and skip\_range), since you might wanna set them to "true" in some cases (for example, for some self-cast spells).

1 ---@type spell\_helper
2 local spell\_helper = require("common/utility/spell\_helper")
3
4 local function can\_cast(local\_player, target)
5 local is\_logic\_allowed = spell\_helper:is\_spell\_castable(spell\_data.id, local\_player, target, false)
6 return is\_logic\_allowed
7 end

Previous

\*\*Spell Book - Raw Functions\*\*

\*\*Next\*

\*\*Graphics - Functions \*\*

\*\*Next\*

\*\*Graphics - Functions \*\*

\*\*Previous Graphics - Functions

Project Sylvanas – 2025

DocsExploreMoreDocumentationHome ♣Discord ♣Roadmap ♣Roadmap ♣