

Minetest Modding Cheatsheet

- <http://dev.minetest.net/>
- <http://www.minetest.net/mods>

General Structure

Each mod goes in its own directory. The directory name is the mod's official name. Items/objects can be referenced by the naming format of “*modname:itemname*”. For instance, the common dirt block's name is “*default:dirt*”. The mod folders follow the general structure seen below. The only required file is “*init.lua*”.

modname

- **depends.txt** - list of mods that this mod needs
- **init.lua** - the main code for the mod
- **textures** - contains block/item images/textures
- **sounds** - sound files (like mob sounds)

Node (Block) Mod

```
minetest.register_node("nuclear_material:radium_ore", {
    description = "Radium Ore",
    tiles = {"default_stone.png^radium_mineral.png"},
    light_source = 2,
    is_ground_content = true,
    groups = {cracky=3},
    drop = 'nuclear_material:radium_lump',
    sounds = default.node_sound_stone_defaults(),
})

minetest.register_craft({
    type = "cooking",
    recipe = "nuclear_material:radium_lump",
    output = "nuclear_material:radium_ore",
})

minetest.register_ore({
    ore_type = "scatter",
    ore = "nuclear_material:radium_ore",
    wherein = "default:stone",
    clust_scarcity = 15*15*15,
    clust_num_ores = 5,
    clust_size = 9,
    height_min = -31000,
    height_max = -10,
})
```

Item or Tool Mod

```
minetest.register_craftitem("nuclear_material:radium_lump", {
    description = "Radium Lump",
    inventory_image = "radium_lump.png",
})

minetest.register_craft({
    type = "fuel",
    recipe = "nuclear_material:radium_lump",
    burntime = "20",
})
```

Crafting Recipe

```
minetest.register_craft({
    output = "modname:item2",
    recipe = {
        {"modname:item", "modname:item"},
        {"modname:item", "modname:item"},
    },
})
```

The above code indicate that “*item2*” can be made by placing four pieces of “*item*” in a square shape anywhere on the crafting grid.

The recipes must be placed in the form of an array that is never larger than three by three.

Attributes and Properties

Groups - properties that determine how an object is broken

- **choppy** - cut with axes
- **cracky** - tough, but breakable with pickaxes
- **crumbly** - broken with shovels
- **explody** - susceptible to explosions
- **fleshy** - damaged by swords
- **oddly_breakable_by_hand** - broken by hand
- **snappy** - snapped with fine tools (i.e. wires and plants)

The groups can be given a value of 1, 2, or 3 to indicate mining speed. “3” is the fastest, so the node/block is broken quickly.

drop - the item that is gained when the object is mined/damaged

output - the item generated by cooking/heating in an oven/furnace

sounds - the sound made when an event occurs

tiles - an array that specifies which image files go on which node face