Custom Items

> The custom item event is a startup event.

Custom items are created in a startup script. They cannot be reloaded without restarting the game. The event is not cancellable.

```
// Listen to item registry event
StartupEvents.registry('item', e => {
  // The texture for this item has to be placed in
kubejs/assets/kubejs/textures/item/test_item.png
  // If you want a custom item model, you can create one in Blockbench and put it in
kubejs/assets/kubejs/models/item/test_item.json
  e.create('test_item')

  // If you want to specify a different texture location you can do that too, like this:
  e.create('test_item_1').texture('mobbo:item/lava') // This texture would be located at
kubejs/assets/mobbo/textures/item/lava.png

  // You can chain builder methods as much as you like
  e.create('test_item_2').maxStackSize(16).glow(true)

  // You can specify item type as 2nd argument in create(), some types have different
available methods
  e.create('custom_sword', 'sword').tier('diamond').attackDamageBaseline(10.0)
})
```

Valid item types:

- basic (this is the default)
- sword
- pickaxe
- axe
- shovel
- shears
- hoe
- helmet

- chestplate
- leggings
- boots

Other methods item builder supports: [you can chain these methods after create()]

- `maxStackSize(size)`
- `displayName(name)`
- `unstackable()`
- `maxDamage(damage)` This is the item's durability, not actual weapon damage.
- `burnTime(ticks)`
- `containerItem(item_id)`
- `rarity('rarity')`
- `tool(type, level)`
- `glow(true/false)`
- `tooltip(text...)`
- `group('group_id')`
- `color(index, colorHex)`
- `texture(customTextureLocation)`
- `parentModel(customParentModelLocation)`
- `food(foodBuilder => ...)` For full syntax see below

Methods available if you use a tool type (`'sword'`, `'pickaxe'`, `'axe'`, `'shovel'` or `'hoe'`):

- `tier('toolTier')`
- `modifyTier(tier => ...)` Same syntax as custom tool tier, see Custom Tiers
- `attackDamageBaseline(damage)` You only want to modify this if you are creating a custom weapon such as Spear, Battleaxe, etc.
- `attackDamageBonus(damage)`
- `speedBaseline(speed)` Same as attackDamageBaseline, only modify for custom weapon types
- `speed(speed)`

Default available tool tiers:

- wood
- stone
- iron
- gold
- diamond
- netherite

Methods available if you use an armour type ('helmet', 'chestplate', 'leggings' or 'boots'):

- `tier('armorTier')`

- `modifyTier(tier => ...)` Same syntax as custom armor tier, see Custom Tiers

Default available armor tiers:

- leather
- chainmail
- iron
- gold
- diamond
- turtle
- netherite

Vanilla group/creative tab IDs:

- search
- buildingBlocks
- decorations
- redstone
- transportation
- misc
- food
- tools
- combat
- brewing

Custom Foods

```
StartupEvents.registry('item', event => {
□event.create('magic_steak').food(food => {
□□food
    □□.hunger(6)
    □□.saturation(6)//This value does not directly translate to saturation points gained
      □□//The real value can be assumed to be:
      □□//min(hunger * saturation * 2 + saturation, foodAmountAfterEating)
      □□.effect('speed', 600, 0, 1)
      □□.removeEffect('poison')
      □□.alwaysEdible()//Like golden apples
      □□.fastToEat()//Like dried kelp
      □□.meat()//Dogs are willing to eat it
      □□.eaten(ctx => {//runs code upon consumption
        □□ctx.player.tell(Text.gold('Yummy Yummy!'))
          □□//If you want to modify this code then you need to restart the game.
```

```
          □□//However, if you make this code call a global startup function
          □□//and place the function *outside* of an event handler
          □□//then you may use the command:
          □□//  /kubejs reload startup_scripts
          □□//to reload the function instantly.
          □□//See example below
        □})
□})

  event.create('magicer_steak').unstackable().food(food => {
    food
      .hunger(7)
      .saturation(7)
      // This references the function below instead of having code directly, so it is
reloadable!
      .eaten(ctx => global.myAwesomeReloadableFunction(ctx))
  })
})

global.myAwesomeReloadableFunction = ctx => {
  ctx.player.tell('Hello there!')
  ctx.player.tell(Text.of('Change me then reload with ').append(Text.red('/kubejs reload
startup_scripts')).append(' to see your changes!'))
}
```

Revision #10
Created 2022-11-28 07:54:22 UTC by ChiefArug
Updated 2023-11-15 06:51:58 UTC by Lexxie