REI Integration

Note: REI integration only works on Fabric in 1.16. In 1.18+, it works on both Forge and Fabric!

> All REI events are client sided and so go in the client_scripts folder

> For 1.19+, see below (this is a temporary page!)

Hide Items

```
onEvent('rei.hide.items', event => {
  event.hide('example:ingredient')
})
```

Add Items

```
onEvent('rei.add.items', event => {
  event.add(Item.of('example:item', { test: 123 })
})
```

Add Information

```
onEvent('rei.information', event => {
  event.add('example:ingredient', 'Title', ['Line 1', 'Line 2'])
})
```

Yeet categories

```
onEvent('rei.remove.categories', event => {
  console.log(event.getCategoryIds()) //log a list of all category ids to
logs/kubejs/client.txt

  //event.remove works too, but yeeting is so much more fun □□
  event.yeet('create:compacting')
})
```

Grouping / Collapsible Entries (1.18.2+)

```
onEvent('rei.group', event => {
  // This event allows you to add custom entry groups to REI, which can be used to clean up the entry list significantly.
  // As a simple example, we can add a 'Swords' group which will contain all (vanilla) swords
  // Note that each group will need an id (ResourceLocation) and a display name (Component / String)
  event.groupItems('kubejs:rei_groups/swords', 'Swords', [
    'minecraft:wooden_sword',
    'minecraft:stone_sword',
    'minecraft:iron_sword',
    'minecraft:diamond_sword',
    'minecraft:golden_sword',
    'minecraft:netherite_sword'
  ])

  // An easy use case for grouping stuff together could be using tags:
  // In this case, we want all the Hanging Signs and Sign Posts from Supplementaries to be grouped together
  event.groupItemsByTag('supplementaries:rei_groups/hanging_signs', 'Hanging Signs', 'supplementaries:hanging_signs')
  event.groupItemsByTag('supplementaries:rei_groups/sign_posts', 'Sign Posts', 'supplementaries:sign_posts')

  // Another example: We want all of these items to be grouped together ignoring NBT,
  // so you don't have a bajillion potions and enchanted books cluttering up REI anymore
  const useNbt = ['potion', 'enchanted_book', 'splash_potion', 'tipped_arrow', 'lingering_potion']

  useNbt.forEach(id => {
    const item = Item.of(id)
    const { namespace, path } = Utils.id(item.id)
    event.groupSameItem(`kubejs:rei_groups/${namespace}/${path}`, item.name, item)
  })

  // Items can also be grouped using anything that can be expressed as an IngredientJS,
  // including for example regular expressions or lists of ingredients
  event.groupItems('kubejs:rei_groups/spawn_eggs', 'Spawn Eggs', [
    /spawn_egg/,
```

```
    /^ars_nouveau:.*_se$/,
    'supplementaries:red_merchant_spawn_egg'
  ])

  // you can even use custom predicates for grouping, like so:
  event.groupItemsIf('kubejs:rei_groups/looting_stuff', 'Stuff with Looting I', item =>
    // this would group together all items that have the Looting I enchantment on them
    item.hasEnchantment('minecraft:looting', 1)
  )

  // you can also group fluids in much the same way as you can group items, for instance:
  event.groupFluidsByTag('kubejs:rei_groups/fluid_tagged_as_water', '\'Water\' (yeah right
lmao)', 'minecraft:water')
})
```

> This below code is meant for 1.19+

Hide Items

```
REIEvents.hide('item', event => {
  event.hide('example:ingredient')
})
```

Add Items

```
REIEvents.add('item', event => {
  event.add(Item.of('example:item', { test: 123 })
})
```

Add Information

```
REIEvents.information(event => {
  event.addItem('example:ingredient', 'Title', ['Line 1', 'Line 2'])
})
```

Yeet categories

```
REIEvents.removeCategories(event => {
  console.log(event.getCategoryIds()) //log a list of all category ids to
logs/kubejs/client.txt
```

```
  //event.remove works too, but yeeting is so much more fun □□
  event.yeet('create:compacting')
})
```

Grouping / Collapsible Entries (1.18.2+)

```
REIEvents.groupEntries(event => {
  // This event allows you to add custom entry groups to REI, which can be used to clean up the entry list significantly.
  // As a simple example, we can add a 'Swords' group which will contain all (vanilla) swords
  // Note that each group will need an id (ResourceLocation) and a display name (Component / String)
  event.groupItems('kubejs:rei_groups/swords', 'Swords', [
    'minecraft:wooden_sword',
    'minecraft:stone_sword',
    'minecraft:iron_sword',
    'minecraft:diamond_sword',
    'minecraft:golden_sword',
    'minecraft:netherite_sword'
  ])

  // An easy use case for grouping stuff together could be using tags:
  // In this case, we want all the Hanging Signs and Sign Posts from Supplementaries to be grouped together
  event.groupItemsByTag('supplementaries:rei_groups/hanging_signs', 'Hanging Signs', 'supplementaries:hanging_signs')
  event.groupItemsByTag('supplementaries:rei_groups/sign_posts', 'Sign Posts', 'supplementaries:sign_posts')

  // Another example: We want all of these items to be grouped together ignoring NBT,
  // so you don't have a bajillion potions and enchanted books cluttering up REI anymore
  const useNbt = ['potion', 'enchanted_book', 'splash_potion', 'tipped_arrow', 'lingering_potion']

  useNbt.forEach(id => {
    const item = Item.of(id)
    const { namespace, path } = Utils.id(item.id)
    event.groupSameItem(`kubejs:rei_groups/${namespace}/${path}`, item.name, item)
  })
```

```
  // Items can also be grouped using anything that can be expressed as an IngredientJS,
  // including for example regular expressions or lists of ingredients
  event.groupItems('kubejs:rei_groups/spawn_eggs', 'Spawn Eggs', [
    /spawn_egg/,
    /^ars_nouveau:.*_se$/,
    'supplementaries:red_merchant_spawn_egg'
  ])

  // you can even use custom predicates for grouping, like so:
  event.groupItemsIf('kubejs:rei_groups/looting_stuff', 'Stuff with Looting I', item =>
    // this would group together all items that have the Looting I enchantment on them
    item.hasEnchantment('minecraft:looting', 1)
  )

  // you can also group fluids in much the same way as you can group items, for instance:
  event.groupFluidsByTag('kubejs:rei_groups/fluid_tagged_as_water', '\'Water\' (yeah right
lmao)', 'minecraft:water')
})
```

Revision #10
Created 2020-09-30 18:04:04 UTC by shedaniel
Updated 2023-01-24 20:22:23 UTC by Max