

Food

Essen für den Spieler und Tiere

Für Spieler: Enums

- Abzählbare Typen ohne Ordnung

- Farben von Gummibärchen
- Lego-Bausteine
- Vogelarten

```
public enum EnumColor {  
    RED,  
    YELLOW,  
    GREEN  
}
```

- Der Name allein definiert den enum (eindeutig)

Eigenes Spieler-Essen

- Neues Essen, das $\frac{1}{2}$ Hunger stillt
- Definition der Essens-Eigenschaften als Enum
 - Enum erweitern, damit er zusätzlich zum Namen Essens-Attribute hat
 - Unter Paket items

```
public enum Food {  
    GINGERBREAD(new FoodProperties.Builder()  
        .nutrition(1).saturationModifier(0.1f)  
        .build());  
  
    private final FoodProperties properties;  
  
    private Food(FoodProperties properties) {  
        this.properties = properties;  
    }  
  
    public FoodProperties get() {  
        return properties;  
    }  
}
```

Neues Item Lebkuchen anlegen

- Erweitere Settings mit food beim Definieren

- In ModItems:

```
public static final Item GINGERBREAD = registerItem("gingerbread",  
    new Properties().food(Food.GINGERBREAD.get()));
```

- Food.GINGERBREAD.get() gibt definierte FoodProperties

- Außerdem: json-Dateien anlegen



```
> mfmd2627 [mfmd2627 main]
> src/main/java
✓ src/main/resources
  > assets.mfmd2627
    > data.minecraft.tags.item
      { } pig_food.json
```

Schweinefutter

```
{
  "replace": false,
  "values": [
    "mfmd2627:gingerbread"
  ]
}
```

Welche Futtertags gibt es?

- armadillo_food, axolotl_food, bee_food, camel_food,
- camel_husk_food, cat_food, chicken_food,
- cow_food (*gilt laut Futterlisten auch für Mooshrooms*),
- fox_food, frog_food, goat_food, happy_ghast_food,
- hoglin_food, horse_food (*gilt auch für Esel und Maultiere*)
- llama_food, nautilus_food, ocelot_food, panda_food,
- parrot_food, pig_food, rabbit_food, sheep_food
- sniffer_food, strider_food, wolf_food, piglin_food

!!! KI-Generiert

Funktioniert alles?

- Git commit und push