

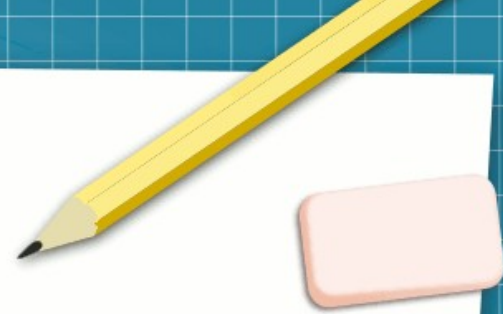


# Tools and Food

Einführung von eigenen Werkzeugen

Einführung von Essen

# Eigenes Werkzeug



- Neue eigene Item-Klasse, die von Item erbt
  - Klasse anlegen
  - Klasse erweitern
  - Klassen verwenden
- Idee: der Stummel soll Blöcke zerschlagen
  - Heuballen zu 9 Weizen
  - getrocknete Seetangblöcke in 9 getrockneten Seetang

# Klasse anlegen

- In de.jottyfan.gta.gdp.item

```
public class StubItem extends Item {  
  
    public StubItem(Settings settings) {  
        super(settings);  
    }  
}
```

# Änderungen in Map registrieren



- Map<Block, Item> als Container für Blöcke, die man mit dem Stummel zerschlagen kann

```
private static final Map<Block, Item> SLASH_MAP = Map.of(  
    Blocks.HAY_BLOCK, Items.WHEAT,  
    Blocks.DRIED_KELP_BLOCK, Items.DRIED_KELP);
```



# Das Zerschlagen programmieren

```
@Override
public ActionResult useOnBlock(ItemUsageContext context) {
    World world = context.getWorld();
    BlockPos pos = context.getBlockPos();
    Block clickedBlock = world.getBlockState(pos).getBlock();
    if (SLASH_MAP.containsKey(clickedBlock)) {
        if (!world.isClient()) {
            world.setBlockState(pos, Blocks.AIR.getDefaultState());
            world.playSound(null, pos, SoundEvents.ITEM_HOE_TILL,
                           SoundCategory.BLOCKS);
            for (int i = 0; i < 9; i++) {
                ItemStack stack = new ItemStack(SLASH_MAP.get(clickedBlock));
                float xScatter = new Random().nextFloat();
                float yScatter = new Random().nextFloat();
                ItemEntity entity = new ItemEntity(world, pos.getX(), pos.getY(), pos.getZ(),
                                                  stack, xScatter, yScatter, 0.2);
                world.spawnEntity(entity);
            }
        }
    }
    return ActionResult.SUCCESS;
}
```

# Item-Registrierung anpassen

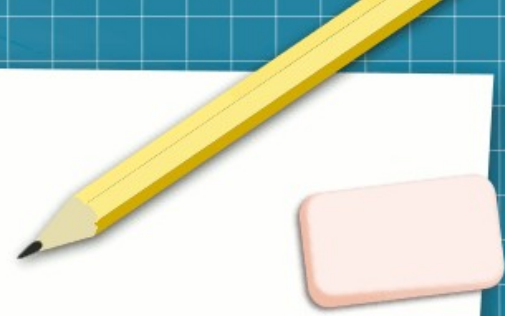


- Durch eigene Klasse ersetzen

```
public class ModItems {  
  
    public static final Item STUB =  
        registerStubItem(Identifier.of(GTAGDP.MOD_ID, "stub"), new Item.Settings());  
  
    private static Item registerStubItem(Identifier identifier, Settings settings) {  
        return Registry.register(Registries.ITEM, identifier,  
            new StubItem(settings.useItemPrefixedTranslationKey().registryKey(  
                RegistryKey.of(RegistryKeys.ITEM, identifier))));  
    }  
}
```

# Arbeit sichern

- Git commit und push



# Eigenes Essen

- Wenn der Stummel gegessen wird, soll  $\frac{1}{2}$  Hunger gestillt werden

```
public class ModFoodComponents {  
  
    public static final FoodComponent DELICIOUS_STUB =  
        new FoodComponent.Builder().nutrition(1)  
            .saturationModifier(0.1f).build();  
}
```



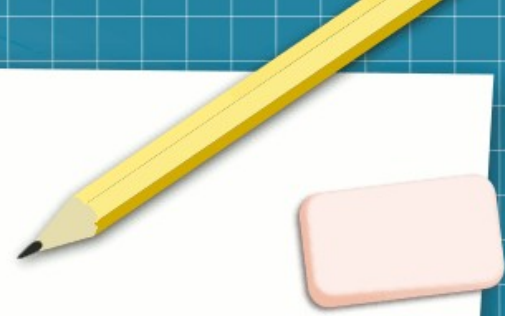
# Stummel zum Essen machen

- In ModItems werden die Settings erweitert

```
public static final Item STUB = registerStubItem(  
    Identifier.of(GTAGDP.MOD_ID, "stub"),  
    new Item.Settings().food(ModFoodComponents.DELICIOUS_STUB));
```

# Arbeit sichern

- Git commit und push



# Herausforderung

- Rezept für Pommes
  - Rezeptkonflikt → geschnittene Kartoffel
  - Icon abmalen
  - Icon für geschnittene Kartoffel malen
  - Icons anlegen
  - Backrezept anlegen
- Ausprobieren und schmecken lassen





This work is licensed under a Creative Commons  
Attribution-ShareAlike 3.0 Unported License.  
It makes use of the works of Mateus Machado Luna.

