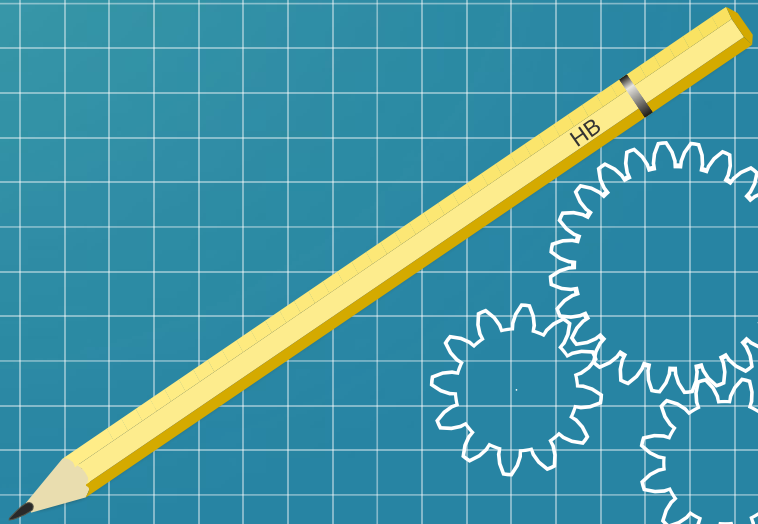
Block States and Item Stacks

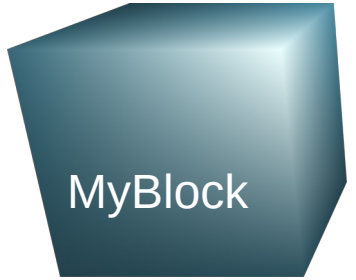
Block States

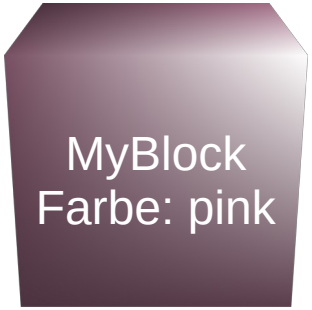
Item Stacks

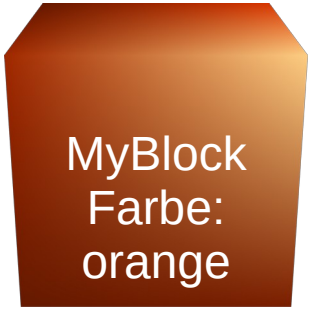


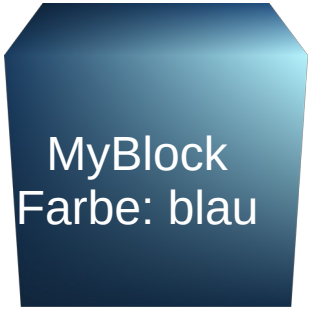
Block States

- Definiert die Eigenschaften einer Block-Instanz
- Ist eindeutig pro Instanz, wird also eher nicht geteilt
- Kann unterschiedliche Eigenschaften halten









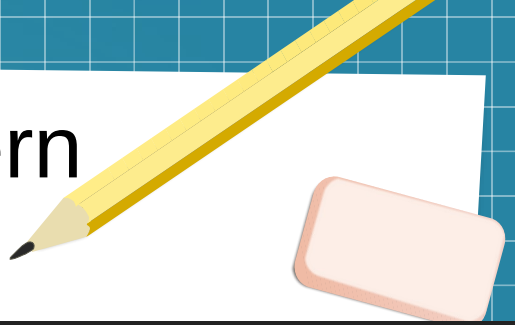
Neue Licht-Property

- Weihnachtsbaumlicht an- und ausschalten
- Property in Klasse definieren: ACTIVATED
- Property registrieren (appendProperties)
- Property benutzen in onUse

Property ACTIVATED definieren

```java
public class ChristmasTree extends SweetBerryBushBlock {
  public static final BooleanProperty ACTIVATED = BooleanProperty.of("activated");

  public ChristmasTree(Settings settings) {
    super(settings);
    setDefaultState(getDefaultState().with(ACTIVATED, true));
  }

  @Override
  protected void appendProperties(Builder<Block, BlockState> builder) {
    builder.add(ACTIVATED);
    super.appendProperties(builder);
  }
```

Property ACTIVATED verändern



```
@Override
protected ActionResult onUse(BlockState state, World world, BlockPos pos, PlayerEntity
  if (state.get(AGE) > 1 && state.get(ACTIVATED)) {
    dropStack(world, pos, new ItemStack(ModItems.GINGERBREAD));
    BlockState blockState = state.with(AGE, 1);
    world.setBlockState(pos, blockState, Block.NOTIFY_LISTENERS);
    world.emitGameEvent(GameEvent.BLOCK_CHANGE, pos, GameEvent.Emitter.of(player, bloc
    return ActionResult.SUCCESS;
  } else if (state.get(AGE)  <= 1) {
    world.setBlockState(pos, state.cycle(ACTIVATED));
    return ActionResult.SUCCESS;
  } else {
    return super.onUse(state, world, pos, player, hit);
  }
}
```

Leuchtkraft für Status verwenden

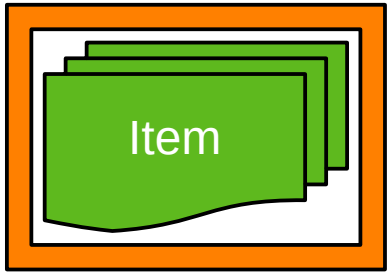
- Leuchtkraft je nach Property ACTIVATED einstellen

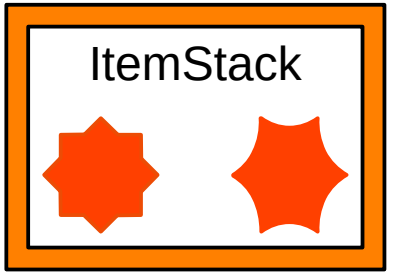
```
public class ModBlocks {

  public static final Block RUBY_BLOCK = registerBlock(Identifier.of(Gtamfmd.MOD_ID, "ruby_block"),
      AbstractBlock.Settings.create().strength(4f).requiresTool().sounds(BlockSoundGroup.AMETHYST_BLOCK));
  public static final Block RUBY_ORE = registerBlock(Identifier.of(Gtamfmd.MOD_ID, "ruby_ore"),
      AbstractBlock.Settings.create().strength(4f).requiresTool());
  public static final Block CHRISTMASTREE = registerSpecialBlock(Identifier.of(Gtamfmd.MOD_ID, "christmastree"),
      AbstractBlock.Settings.create().ticksRandomly().noCollision().luminance(state -> state.get(ChristmasTree.ACTIVATED) ? 15 : 0),
      x -> new ChristmasTree(x));
```

```
.luminance(state -> state.get(ChristmasTree.ACTIVATED) ? 15 : 0),
```

Item Stacks

- Sammlung von gleichen Items mit gleichen Eigenschaften
- Kann in Hand gehalten oder Kisten gelegt werden
- Hat eine Stacksize, normalerweise maximal 64





Lebenszeiten von Items und ItemStacks

- Item überlebt Neustart nicht
- ItemStack hat Attribute, die den Neustart überleben
  - Damage-Attribut
  - Verzauberung
  - Einfärbung (Bett, Wolle)
- Weiterführende Details: Einsatz von nbt
- Siehe https://youtu.be/2FotyGYMGUU?feature=shared