

24 Přidání funkčnosti itemu anti_creeper

V této lekci si do hry přidáme model itemu, který jsem si tvořili v předchozí lekci a poté začneme pracovat na funkčnosti, aby item zabránil výbuchu creepera v okolí hráče. Item budeme přidávat do módu `custom-item`, který jsme si tvořili v počátečních lekcích.

Přidání modelu a textury

Přidání modelu a textur funguje stejně jako u bloku. Z `blockbench` bychom měli mít stažený `.zip` soubor, ve kterém máme model (`.json` soubor) a textury (`.png` soubory).

Model přesuneme do složky `src/main/resources/assets/custom-item/models/item`.
Textury přesuneme do složky `src/main/resources/assets/custom-item/textures/item`.

Abychom si mohli u itemu změnit jeho funkčnost, tak si musíme vytvořit vlastní třídu. Tu si vytvoříme ve složce `src/main/java/com/example/item` a nazveme ji `AntiCreeperItem`. Do třídy doplníme základní strukturu pro item.

`src/main/java/com/example/item/AntiCreeperItem.java`

```
1 public class AntiCreeperItem extends Item {  
2     public AntiCreeperItem(Settings settings) {  
3         super(settings);  
4     }  
5 }
```

Vytvoření třídy

Třidu vytvoříme tak, že klikneme pravým tlačítkem myši na složku, kde chceme třídu vytvořit. Dále zvolíme `New > Java Class`.

Dále se přesuneme do souboru `ModItems`, kde jsme si přidávaly itemy a následujícím kódem si náš item registrujeme. Zároveň si můžeme nastavit například omezení na 1 item ve stacku.

`src/main/java/com/example/item/ModItems.java`

```
1 public static final Item ANTI_CREEPER_ITEM = register(  
2     new AntiCreeperItem(new Item.Settings().maxCount(1)),  
3     "anti_creeper"  
4 );
```

Poznámka

Abychom si mohli k itemu doplnit funkčnost, tak zde musíme použít naši nově vytvořenou třídu `AntiCreeperItem`

Nyní bychom měli mít item i s texturou a modelem ve hře. Můžeme jej ale zatím získat jen pomocí příkazu `give`

Přidání funkčnosti itemu

Abychom upravili, co se děje, když má hráč item v inventáři, tak můžeme použít metodu `inventoryTick`. Stačí, když ve třídě `AntiCreeperItem` začnete psát `inv` a v nápovědě by se měla nabídnout metoda `inventoryTick`.

```
src/main/java/com/example/item/AntiCreeperItem.java
```

```
1  @Override
2  public void inventoryTick(ItemStack stack, World world, Entity entity, int
    slot, boolean selected) {
3      super.inventoryTick(stack, world, entity, slot, selected);
4  }
```

Jak už název napovídá, tak tato metoda se spouští každý tick (20krát za sekundu) a to jen v případě, že má hráč item v inventáři.

Samotný postup pro zabránění výbuchu creepera bude následující:

1. zkontrolujeme, že entita, která item drží je hráč
2. zjistíme zda hráč item drží v ruce
3. vyhledáme creepery v okolí hráče
4. resetujeme creeperům čas do výbuchu

Kontrolu, zda item drží hráč provedeme pomocí metody `isPlayer`. Kontrolu můžeme otočit tím, že před podmínku napíšeme `!`. Pokud tedy entita **není** hráč, tak dál nemusíme pokračovat a můžeme pomocí `return` kód ukončit. Metoda vrací hodnotu `void`, což je doslova *nic*, takže za `return` žádnou hodnotu psát nemusíme.

```
src/main/java/com/example/item/AntiCreeperItem.java
```

```
1  @Override
2  public void inventoryTick(ItemStack stack, World world, Entity entity, int
    slot, boolean selected) {
3      if (!entity.isPlayer()){
4          return;
5      }
6
7      super.inventoryTick(stack, world, entity, slot, selected);
```

 Tip

Abychom nemuseli vždy u proměnné psát typ, tak můžeme použít slovo `var` a typ se nastaví automaticky.

Dále si entitu převedeme z obecné entity na typ `PlayerEntity` (hráč).

```
1 var player = (PlayerEntity)entity;
```

Pomocí metody `getHandItems` si získáme seznam itemů, které má hráč v ruce. Dále si seznam převedeme na jiný typ pomocí metody `iterator`.

```
1 var items = player.getHandItems().iterator();
```

Abychom mohli zkontrolovat, že je aktuální stack itemů v jedné nebo druhé ruce, tak jej musíme převést na jiný typ seznamu pomocí `Lists.newArrayList(items)` a dále pak můžeme použít metodu `contains`, která zkontroluje, zda daný seznam obsahuje konkrétní prvek. V našem případě je to stack itemů v parametru `stack`. Výsledek si uložíme do proměnné `isItemInHand`

```
1 var isItemInHand = Lists.newArrayList(items).contains(stack);
```

Pokud hráč item nemá v ruce, tak můžeme opět kód ukončit pomocí `return`.

```
1 if (!isItemInHand){
2     return;
3 }
```

Celý kód bude vypadat následovně:

src/main/java/com/example/item/AntiCreeperItem.java

```
1 @Override
2 public void inventoryTick(ItemStack stack, World world, Entity entity, int
   slot, boolean selected) {
3     if (!entity.isPlayer()){
4         return;
5     }
6
7     var player = (PlayerEntity)entity;
8     var items = player.getHandItems().iterator();
9     var isItemInHand = Lists.newArrayList(items).contains(stack);
```

```
10      if (!isItemInHand){
11          return;
12      }
13
14      super.inventoryTick(stack, world, entity, slot, selected);
15  }
```

Zbytek kódu si doděláme v následujících lekcích.