

18 Vlastní textura bloku

V této lekci si ukážeme, jak si vytvořit vlastní texturu bloku a jak ji přidat do našeho módu. Na to budeme používat program Blockbench, ve kterém jsme si tvořili texturu hůlky ([12-tvorba-textury-hulky](#)). Hůlku jsme tvořili jen jako 2D obrázek, ale texturu bloku musíme tvořit ve 3D, abychom věděli, jak potom blok bude vypadat ve hře.

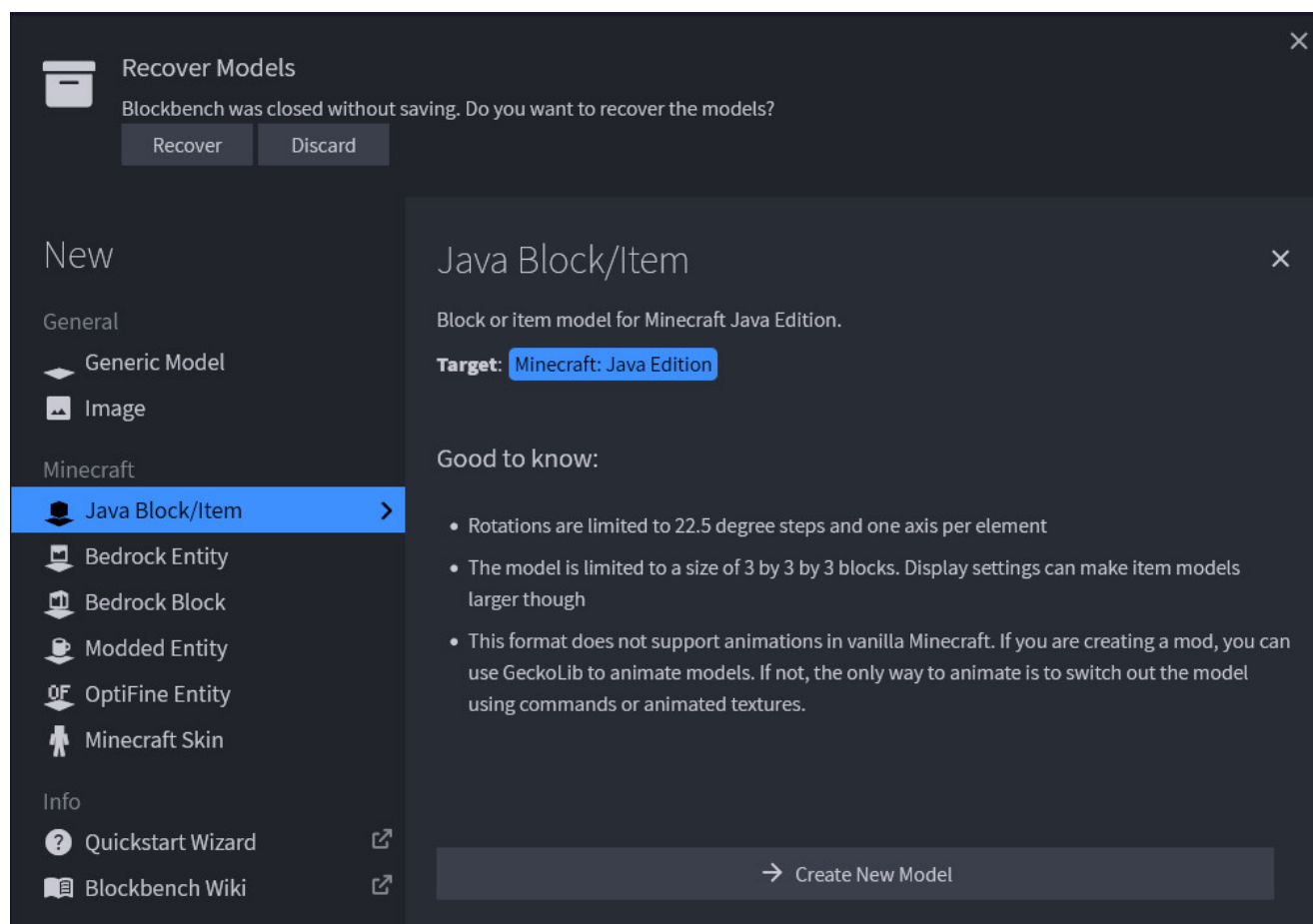
Tvorba textury bloku

Blockbench je program na 3D jednoduché modelování a vytváření textur na 3D modely. Zároveň přímo podporuje tvorbu modelů a textur do Minecraftu a na některé nové stvoření je používají i samotní vývojáři Minecraftu.

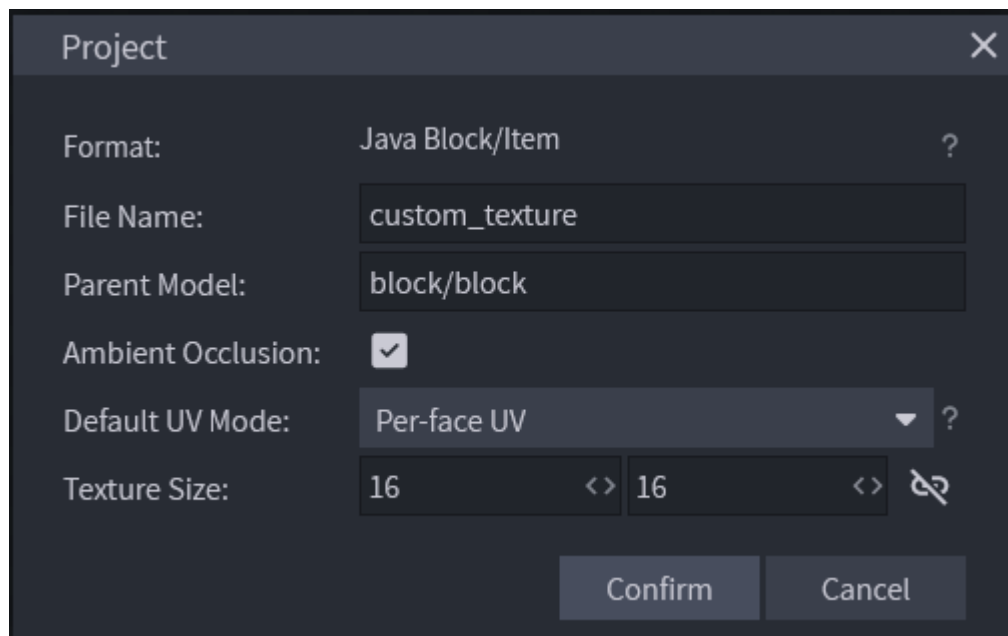
Program a informace k němu najdeme na blockbench.net. Výhodou tohoto programu je to, že funguje i v prohlížeči a není nutné jej stahovat a instalovat. My budeme prozatím používat právě webovou verzi tohoto programu, která je dostupná na web.blockbench.net.

Vytvoření projektu

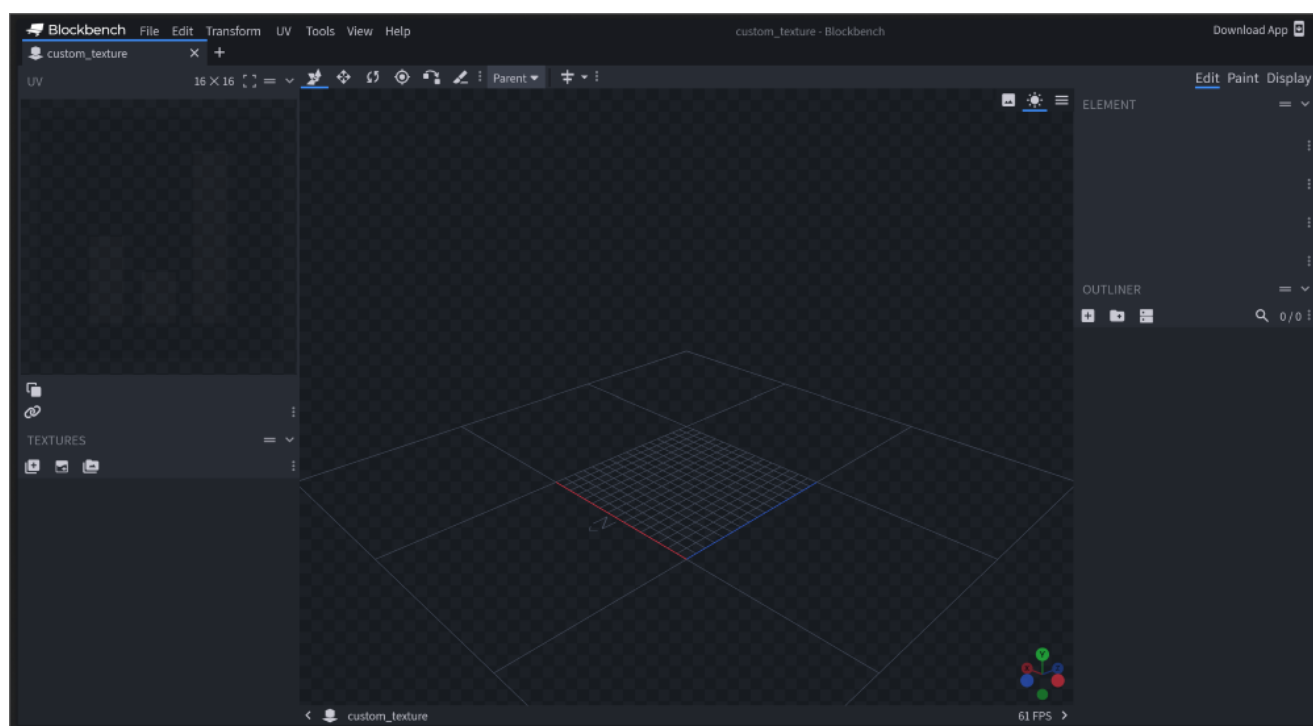
Po otevření stránky vidíme, jaké všechny projekty můžeme vytvořit. My si zvolíme **Java Block/Item**.



Dále se nám zobrazí okno pro vytvoření projektu, kde si zadáme hlavně název projektu/bloku. Zatím nebudeme tvořit žádný konkrétní blok, takže jej můžeme nazvat klidně jen `custom_texture`. Dále jako *Parent Model* zvolíme `block/block`. Tím zaručíme, že náš blok bude vypadat stejně jako všechny ostatní bloky v Minecraftu - například velikost bloku v inventáři, v ruce atd.



Okno editoru vypadá následovně:



Ovládání

Uprostřed máme plochu, kde bude náš 3D model. Základní ovládání je následující:

- levé tlačítko myši - otáčení
- pravé tlačítko myši - posouvání

- kolečko myši - přiblížení/oddálení

Outliner

Na pravé straně vidíme část, kde se nám budou zobrazovat jednotlivé části našeho modelu. Ikonou **+** můžeme přidat nový model. A protože jsme si zvolili Minecraft projekt, tak nemůžeme přidat nic jiného, než krychli.

Element

Vpravo nahoře vidíme nastavení zvoleného objektu. Můžeme nastavovat:

- pozici - *position*
- velikost - *size*
- bod otáčení - *pivot point*
- rotaci - *rotation*

Základní blok v Minecraftu má velikost **16x16x16**, proto si změníme velikost právě na tyto hodnoty.

Textures

Vlevo vidíme část, kde se nám zobrazují textury. Pomocí tlačítka vpravo vytvoříme novou texturu (po přejetí myši se zobrazí nápis *Create Texture*). Jako název opět zvolíme **custom_texture** a zbytek můžeme nechat v základním nastavení.

Create Texture

Name:

custom_texture

Folder:

Type:

Texture Template

Solid Color Template

Blank

Pixel Density:

16x

?

Color:

Rearrange UV:

☒

?

Box UV:

☐

Power-of-2 Size:

☒

?

Keep Multi Texture Occupancy:

☒

?

Padding:

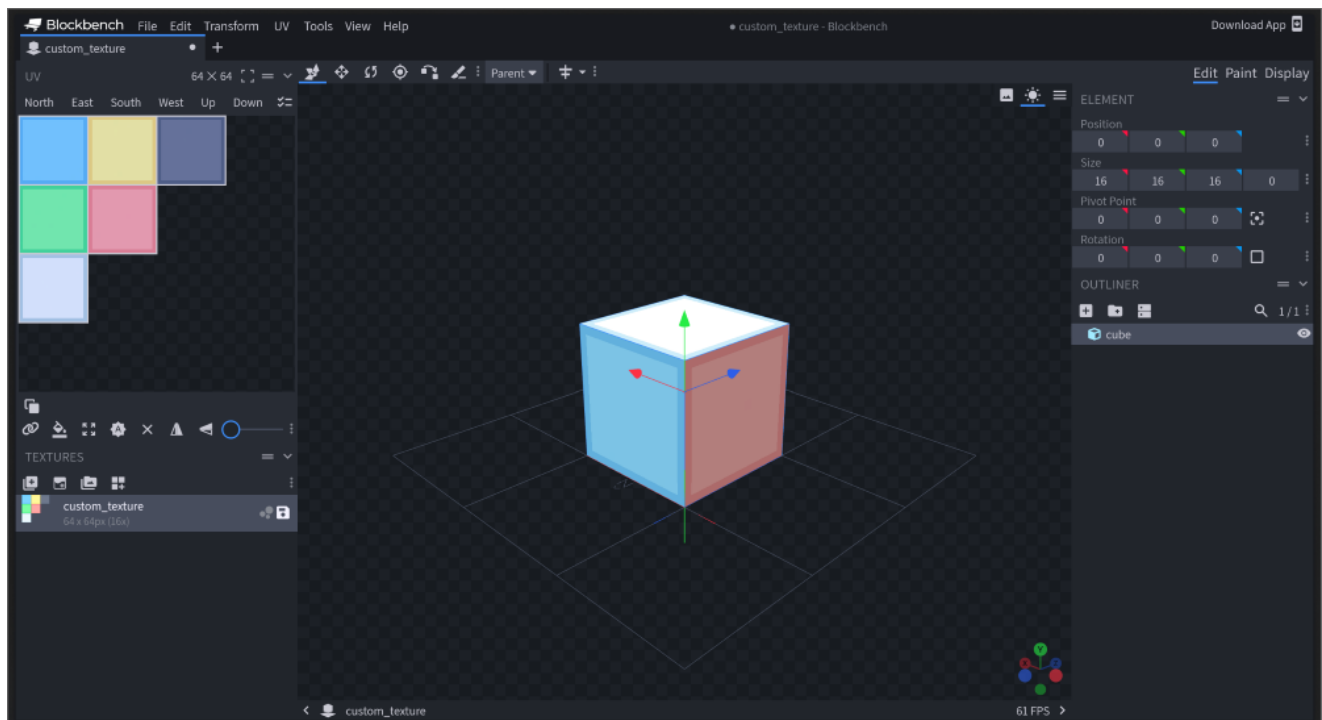
☐

?

Confirm

Cancel

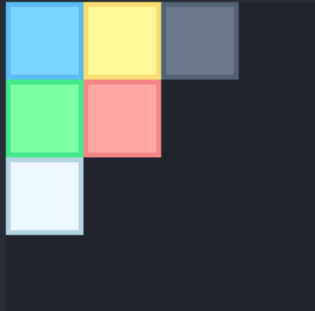
Nyní bychom měli mít takovouto barevnou kostku:



U textury je dále potřeba zvolit namespace projektu, aby se ve hře textura správně načetla. To uděláme tak, že na texturu klikneme pravým tlačítkem myši a zvolíme *Properties*. Otevře se nám okno, kde máme kromě dalšího nastavení textury právě i možnost nastavit

namespace. Pro náš projekt si zvolíme namespace `custom-block`. Jinak bychom si zvolili namespace podle módu, do kterého blok tvoříme.

custom_texture (64 x 64)



Name:

custom_texture

Variable:

0

Folder:

block

Namespace:

custom-block

Render Mode:

Default

Frame Time:

1

<> ?

Interpolate:

☐

?

Loop Mode:

Loop

Confirm

Cancel

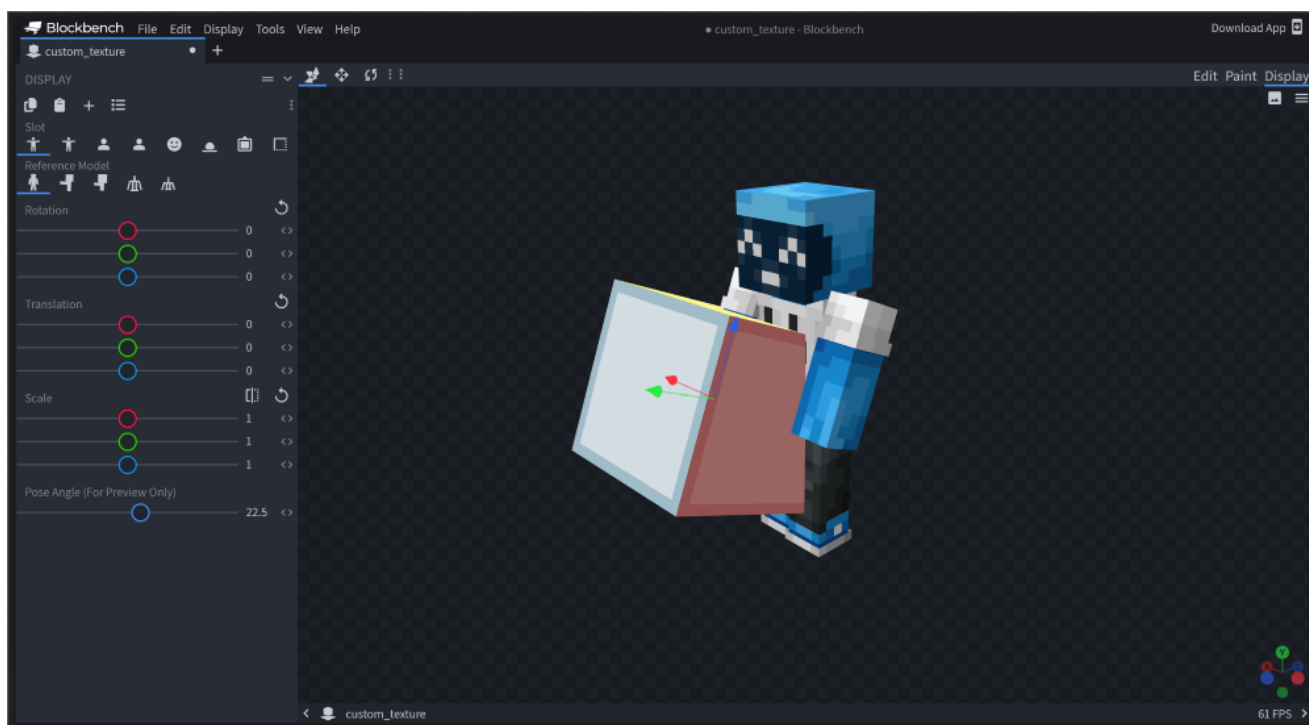
Strany

Takto by kostka vypadala i po vložení do hry. Při tvoření je důležité vědět, která strana se bude kde zobrazovat ve hře. Nejdůležitější je asi světle modrá strana, kde dole vidíme N s šipkou. Tato strana se totiž bude zobrazovat vepředu. Pokud tedy hráč blok položí, tak tato strana bude směřovat k němu. Například blok pece tam má otvor, kde se poté zobrazuje, že je pec aktivní.

Zobrazení

Abychom si dokázali představit, jak bude blok vypadat ve hře, tak zde máme záložku na zobrazení bloku/itemu v různých situacích. To najdeme v pravém horním rohu - **Display**. V

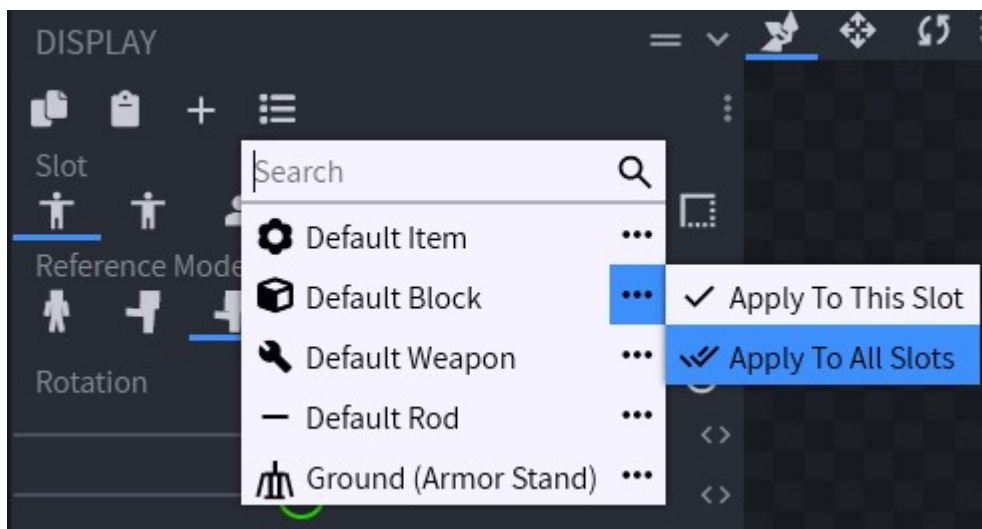
základu vypadá blok následovně:



Zde si můžeme nastavit, jak se bude blok zobrazovat na různých místech ve hře. U každého místa si můžeme zvolit rotaci, posunutí a velikost. Tyto hodnoty si můžeme pro každé místo nastavit zvlášť.

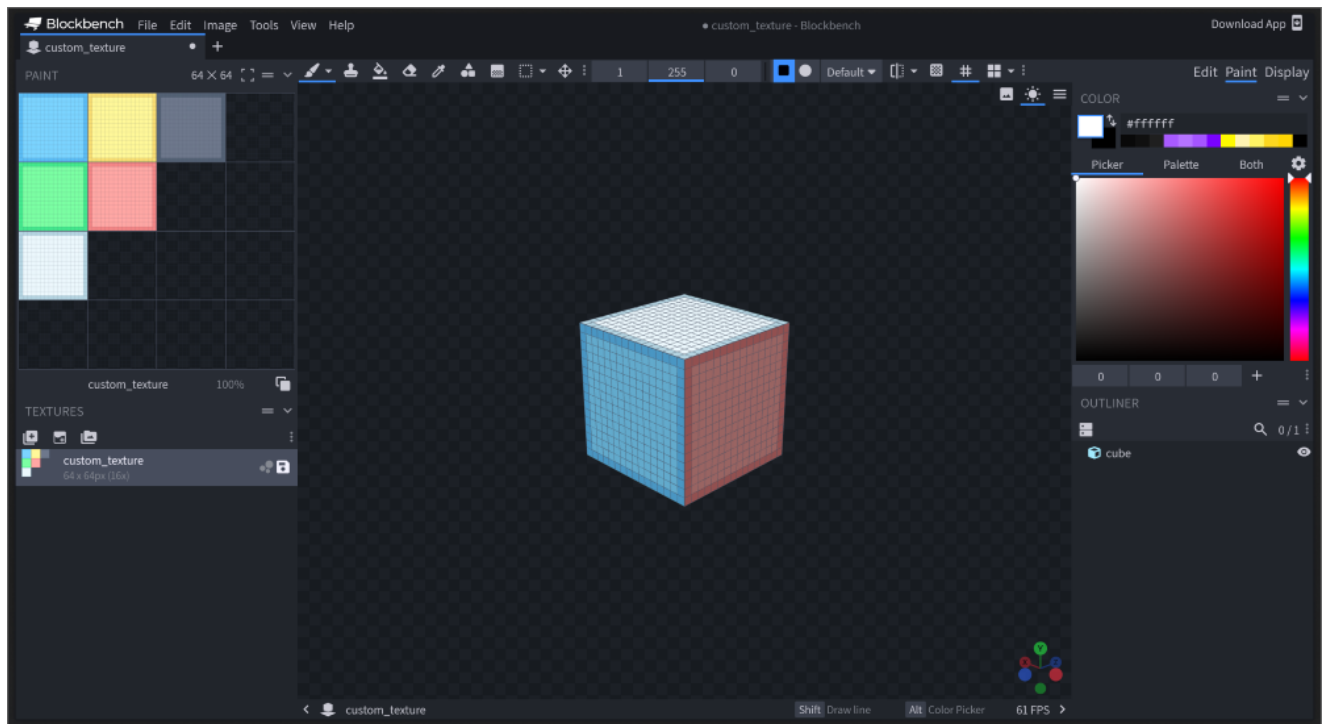
Tím, že jsme ale při vytváření uvedli, že má blok vycházet z nastavení `block/block`, tak by se takto reálně blok nezobrazoval a zobrazoval by se stejně jako všechny ostatní bloky ve hře. Blockbench ale nemůže vědět, jaké nastavení je v souboru `block/block`. My si ale můžeme nastavení změnit.

Pokud chceme nastavit zobrazování, tak jak je v Minecraftu v základu, tak můžeme použít předvolbu (*Preset*) pro blok.



Malování

Jako poslední si ukážeme záložku **Paint**, kde upravujeme samotnou texturu.

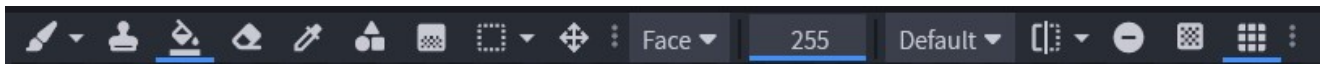


V pravé části si můžeme měnit barvy. Máme možnost si vybírat buď jakoukoliv barvu (*Picker*) nebo z připravené palety barev (*Palette*).

V horní liště najdeme nástroje a jejich nastavení.



Nás bude zajímat hlavně štětec - ten používáme na malování. Pomocí prvního čísla na liště si můžeme měnit velikost štětce.

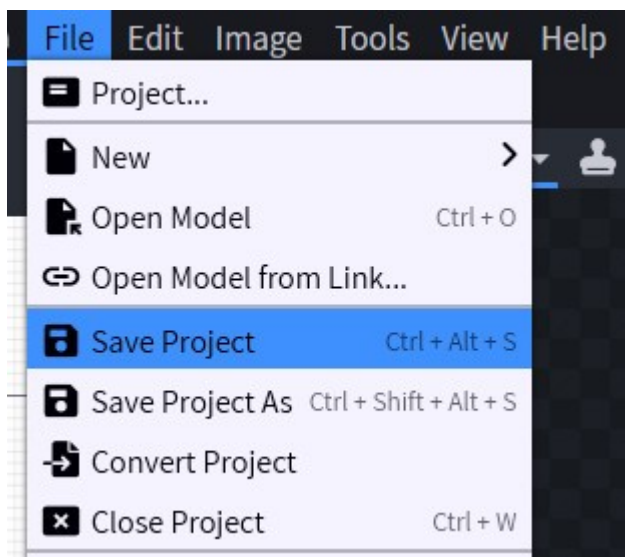


Poté zde máme kyblík, který používáme na vyplnění větší oblasti barvou. Zde si můžeme nastavit, co má kyblík vyplnit. V základu vyplňuje celou jednu stěnu objektu. Máme zde možnosti:

- Face - celá stěna objektu - základní nastavení
- Element - celý objekt
- Connected Colors - nahraní barvu, na kterou klikneme, ale pouze v oblasti, kam jsme kliknuli
- Colors - nahradí barvu, na kterou klikneme na celé stěně objektu

Uložení projektu

Projekt je dobré si průběžně ukládat. To uděláme v záložce *File* a poté zvolíme *Save Project*.



Projekt se ukládá v souboru typu `bbmodel` (blockbench model). Tento soubor poté může otevřít tak, že zvolíme *File* → *Open Model*.

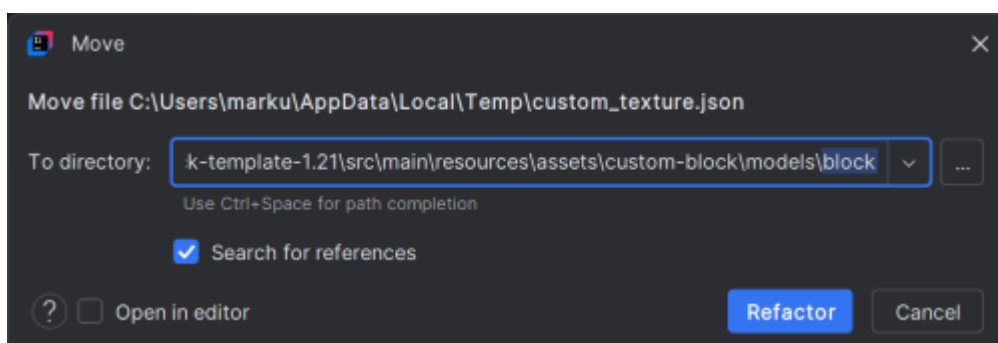
Pokud chceme blok vložit do Minecraftu, tak potřebujeme `json` soubor, kde bude model bloku a `png` soubor, kde bude textura. Tyto dva soubory získáme tak, že zvolíme *File* → *Download Archive*. Tím se nám uloží `zip` soubor, kde budou potřebné soubory.

Přidání textury do hry

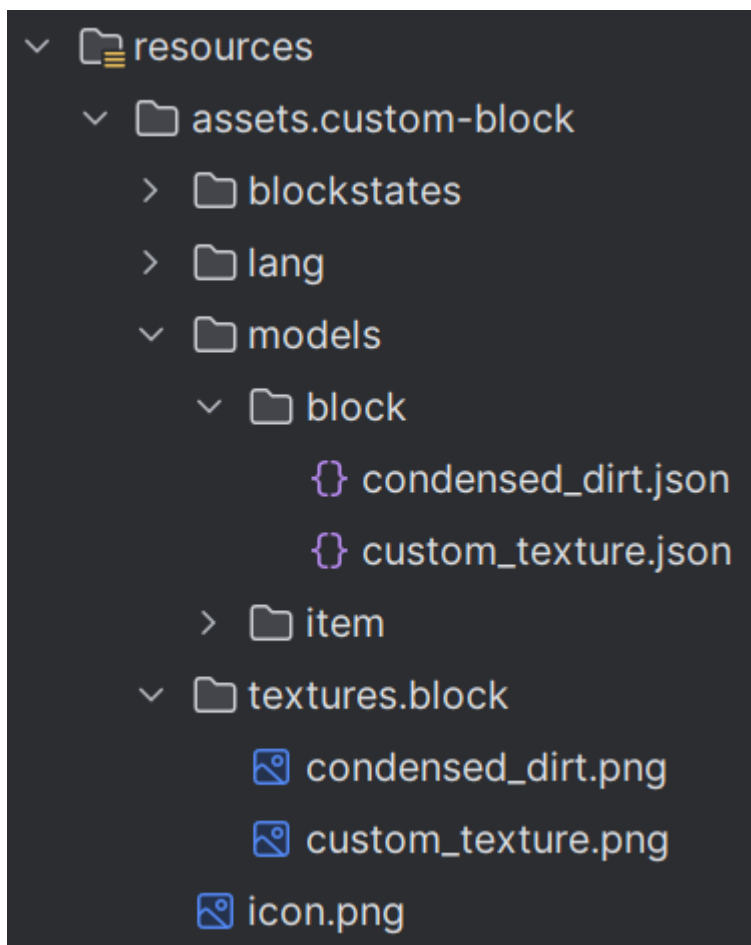
Modely (`json` soubory) budeme vkládat do složky `src/main/resources/assets/custom-block/models/block`.

Textury (`png` soubory) budeme vkládat do složky `src/main/resources/assets/custom-block/textures/block`.

Pro vložení bloku tedy stačí otevřít stažený `zip` soubor z Blockbench, kde máme uložený model a texturu bloku. A poté přesunout soubory do příslušných složek. Mělo by se nám zobrazit okno s informací, jaký soubor kam kopírujeme. Tam stačí zvolit *Refactor*. Stejně vložíme i druhý soubor.



Výsledek by měl vypadat následovně:



Nastavení bloku

Aby se nám blok ve hře zobrazoval, tak potřebujeme nastavit několik věcí:

Blockstate

Jaký model se má v základu použít pro zobrazení bloku.

Ve složce `src/main/resources/assets/custom-block/blockstates` vytvoříme soubor `custom_texture.json`.

Obsah souboru bude vypadat následovně:

src/main/resources/assets/custom-block/blockstates/custom_texture.json

```
1  {
2    "variants": {
3      "": {
4        "model": "custom-block:block/custom_texture"
5      }
6    }
7  }
```

Model itemu

Jak má vypadat item bloku (pokud item přidáváme).

Ve složce `src/main/resources/assets/custom-block/models/item` vytvoříme soubor `custom_texture.json`.

Obsah souboru bude vypadat následovně:

```
src/main/resources/assets/custom-block/models/item/custom_texture.json
```

```
1  {
2    "parent": "custom-block:block/custom_texture"
3  }
```

Přidání bloku do hry

Jako poslední krok ještě musíme v kódu pro mód registrovat blok do hry. To uděláme v souboru `ModBlocks.java`. Toto už jsme dělali v lekcí [13 Přidání bloku - Vytvoření a nastavení bloku](#)

Musíme přidat klíč, pod kterým chceme blok (a item) registrovat. To je zároveň id bloku/itemu.

```
src/main/java/com/example/block/ModBlocks.java
```

```
1  public static final RegistryKey<Block> CUSTOM_TEXTURE_KEY =
    RegistryKey.of(
2      RegistryKeys.BLOCK,
3      Identifier.of(CustomBlock.MOD_ID, "custom-texture")
4  );
```

Dále registrujeme samotný blok a případně změníme jeho nastavení.

```
src/main/java/com/example/block/ModBlocks.java
```

```
1  public static final Block CUSTOM_TEXTURE = register(
2      new Block(AbstractBlock.Settings.create()),
3      CUSTOM_TEXTURE_KEY,
4      true
5  );
```

Nyní bychom ve hře měli mít blok i item s texturou, kterou jsme si vytvořili.