Appendix B: The Code, Typed Out

Every file in the finished sword, exactly as it is in the Starter Kit.

There are three ways to get these files into your packs:

1. **Download the Starter Kit** (the easiest). The files are ready to copy.
2. **Type them** into VS Code. It's slow, but you'll read every line.
3. **Photograph a page and copy the text** with a text-scanning app, then paste it into VS Code:
   - iPhone or iPad: point the Camera at the page, tap the text icon (Live Text).
   - Android: Google Lens, then **Text**.
   - Windows: Snipping Tool's **Text actions**, or PowerToys Text Extractor.

Check what the app copied. Scanning apps sometimes mix up `0` and `O`, `1` and `l`, and turn straight quotes `"` into curly ones. In our test the trickiest were a `0` on its own line and the backticks `` ` `` in the last message line of **main.js**. VS Code helps: anything wrong turns red or shows in the Problems count.

Spaces at the start of a line only make the file easier to read. Minecraft doesn't mind if they're missing. VS Code can tidy them: right-click, **Format Document**.

The pages are sideways so that no line has to wrap. Turn the book, or turn the photo.

Not here: the picture (`my_sword.png`), the battle cry (`my_sword_hit.ogg`) and the two `pack_icon.png` files. You make those yourself in Chapters 5 and 7, or copy them from the kit.

Change these to yours: the UUIDs (from your worksheet), `family:my_sword` (your sword's ID), `My Sword` (its name) and `my_sword_owner` (your owner tag).

MySword_RP \ manifest.json

Resource pack name tag. Put your own UUIDs (worksheet lines 1 and 2) where shown. Chapter 4.

```
{
  "format_version": 2,
  "header": {
    "name": "My Sword RP",
    "description": "My sword's looks and sounds",
    "uuid": "556befb8-d2ca-4641-9bc1-e1da82b4fff1",
    "version": [
      1,
      0,
      0
    ],
    "min_engine_version": [
      1,
      26,
      0
    ]
  },
  "modules": [
    {
      "type": "resources",
      "uuid": "60cda337-25e6-4dea-bbd9-9fe720479a02",
      "version": [
        1,
        0,
        0
      ]
    }
  ]
}
```

MySword_RP \ textures \ item_texture.json

Which picture belongs to the sword. Chapter 6.

```
{
  "resource_pack_name": "My Sword RP",
  "texture_name": "atlas.items",
  "texture_data": {
    "family:my_sword": {
      "textures": "textures/items/my_sword"
    }
  }
}
```

MySword_RP \ sounds \ sound_definitions.json

Names the battle cry sound. Chapter 7.

```
{
  "format_version": "1.14.0",
  "sound_definitions": {
    "my_sword.hit": {
      "category": "player",
      "sounds": [
        "sounds/my_sword/my_sword_hit"
      ]
    }
  }
}
```

MySword_BP \ manifest.json (part 1 of 2)

Behavior pack name tag. Your UUIDs: lines 3, 4, 5, and a copy of line 1. Chapter 8.

```
{
  "format_version": 2,
  "header": {
    "name": "My Sword BP",
    "description": "My sword: the item, its stats, recipe and power",
    "uuid": "36f9aae8-40ea-47c1-8d8e-88861a71fec1",
    "version": [
      1,
      0,
      0
    ],
    "min_engine_version": [
      1,
      26,
      0
    ]
  },
  "modules": [
    {
      "type": "data",
      "uuid": "3b823d31-68db-4651-9802-ed46a5abe5f0",
      "version": [
        1,
        0,
        0
      ]
    },
```

MySword_BP \ manifest.json (part 2 of 2)

Continued: type it straight after the previous part.

```
    {
      "type": "script",
      "language": "javascript",
      "uuid": "e4b16501-a502-46b7-853d-825f5982b6fd",
      "version": [
        1,
        0,
        0
      ],
      "entry": "scripts/main.js"
    }
  ],
  "dependencies": [
    {
      "uuid": "556befb8-d2ca-4641-9bc1-e1da82b4fff1",
      "version": [
        1,
        0,
        0
      ]
    },
    {
      "module_name": "@minecraft/server",
      "version": "2.11.0"
    }
  ]
}
```

MySword_BP \ items \ my_sword.json (part 1 of 2)

The sword itself: ID, name, picture and stats. Chapter 7.

```
{
  "format_version": "1.26.60",
  "minecraft:item": {
    "description": {
      "identifier": "family:my_sword",
      "menu_category": {
        "category": "equipment",
        "group": "minecraft:itemGroup.name.sword"
      }
    },
    "components": {
      "minecraft:icon": "family:my_sword",
      "minecraft:display_name": { "value": "My Sword" },
      "minecraft:max_stack_size": 1,
      "minecraft:hand_equipped": true,
      "minecraft:can_destroy_in_creative": false,
```

MySword_BP \ items \ my_sword.json (part 2 of 2)

Continued: type it straight after the previous part.

```
      "minecraft:damage": 12,
      "minecraft:durability": { "max_durability": 100 },
      "minecraft:enchantable": { "slot": "sword", "value": 10 },
      "minecraft:repairable": {
        "repair_items": [
          { "items": ["minecraft:diamond"], "repair_amount": 25 }
        ]
      },

      "minecraft:glint": true,
      "minecraft:rarity": "epic",
      "minecraft:fire_resistant": { "value": true },
      "minecraft:swing_sounds": {
        "attack_hit": "my_sword.hit",
        "attack_critical_hit": "my_sword.hit"
      }
    }
  }
}
```

MySword_BP \ recipes \ my_sword.json

How to craft it. Chapter 10.

```
{
  "format_version": "1.20.10",
  "minecraft:recipe_shaped": {
    "description": {
      "identifier": "family:my_sword"
    },
    "tags": ["crafting_table"],
    "pattern": [
      "D",
      "B",
      "#"
    ],
    "key": {
      "D": { "item": "minecraft:diamond" },
      "B": { "item": "minecraft:blaze_powder" },
      "#": { "item": "minecraft:stick" }
    },
    "unlock": [
      { "item": "minecraft:diamond" }
    ],
    "result": {
      "item": "family:my_sword"
    }
  }
}
```

MySword_BP \ scripts \ main.js (part 1 of 3)

The special power. Chapter 10.

```
// My Sword script.
// Lines starting with // are notes for people. Minecraft ignores them.

import { world, EquipmentSlot, Player } from "@minecraft/server";

// ---- Settings: easy things to change ----

// Your sword's ID (the same as in the item file).
const SWORD_ID = "family:my_sword";

// The tag a player needs before the sword's power works for them.
// Give it with:   /tag @s add my_sword_owner
// Take it away:   /tag @s remove my_sword_owner
const OWNER_TAG = "my_sword_owner";

// How many seconds the target burns.
const FIRE_SECONDS = 5;

// The power works on 1 hit in this many. 4 means 1 hit in 4. 1 means every hit.
const POWER_CHANCE = 4;

// How many flame sparks appear when the power works.
const FLAME_COUNT = 12;
```

MySword_BP \ scripts \ main.js (part 2 of 3)

Continued: type it straight after the previous part.

```
// ---- The sword's code ----

// Minecraft runs this every time one creature hits another.
world.afterEvents.entityHitEntity.subscribe((event) => {
  const attacker = event.damagingEntity; // who swung
  const target = event.hitEntity; // who got hit

  // Only players hold swords we care about.
  if (!(attacker instanceof Player)) return;

  // What is the player holding in their main hand?
  const equipment = attacker.getComponent("minecraft:equippable");
  const item = equipment?.getEquipment(EquipmentSlot.Mainhand);

  // Only react to our sword.
  if (item?.typeId !== SWORD_ID) return;

  // Is this player the sword's owner?
  if (!attacker.hasTag(OWNER_TAG)) {
    // Not the owner: a normal hit, no power, and a message just for them.
    attacker.sendMessage("This sword only obeys its owner!");
    return;
  }

  // The power only works on mobs, not on other players.
  if (target.typeId === "minecraft:player") return;
```

MySword_BP \ scripts \ main.js (part 3 of 3)

Continued: type it straight after the previous part.

```
  // Roll the dice: only 1 hit in POWER_CHANCE gets the power.
  if (Math.random() * POWER_CHANCE >= 1) return;

  // Turn "minecraft:husk" into "husk".
  const targetName = target.typeId.replace("minecraft:", "");

  // The special power: set the target on fire.
  target.setOnFire(FIRE_SECONDS, true);

  // A whoosh sound, like flint and steel.
  target.dimension.playSound("fire.ignite", target.location);

  // A burst of flame sparks around the target's body.
  for (let i = 0; i < FLAME_COUNT; i++) {
    target.dimension.spawnParticle("minecraft:basic_flame_particle", {
      x: target.location.x + (Math.random() - 0.5),
      y: target.location.y + Math.random() * 2,
      z: target.location.z + (Math.random() - 0.5),
    });
  }

  // A message so we can see the power worked.
  attacker.sendMessage(`Zap! The ${targetName} is on fire!`);
});
```