Amethyst Algorithms Project 1: Coding Basics

Henry Post

Drakonix Systems, LLC

April 13, 2025

About the author

Henry Post

`bio  -> meltingscales.github.io`
`code -> github.com/meltingscales/amethyst-algorithms`

This presentation was made using Markdown and Pandoc with the “beamer” template.

What is this?

Amethyst Algorithms is a course aimed at people aged 15 or older, meant to teach you how to do software programming using Minecraft and NeoForged.

If you're aged 15 or older, like Minecraft, and want to learn programming, you should give this course a try.

Tooling setup

You'll need to install a few tools first!

If using Windows, I highly recommend using `winget` to install packages. It is a fantastic package manager similar to `apt` or `yum`.

- winget

If using Linux, this will depend. I recommend Ubuntu for beginners, but I personally use NixOS.

1. Install IntelliJ IDEA
   - Windows JetBrains Toolbox
   - Linux/OSX JetBrains Toolbox
2. Install Git
   - Git on Windows
   - Git on Linux
   - Git on OSX
3. Install Java Development Kit 21
   - JDK 21 on Windows
   - JDK 21 on Linux
   - JDK 21 on OSX

Git setup

I'll assume you've installed Git. You just need to run the below:

1. Open a terminal and `cd` to the place you'd like the repository to live.
2. Run `git clone https://github.com/meltingscales/amethyst-algorithms-minecraft-intro-to-cs-course`

```
> INSTRUCTOR: Clone the repo fresh.
```

Java setup

Just install JDK 21 (not JRE 21).

```
> INSTRUCTOR: Install JDK 21.
```

IDE setup

Next, we're going to set up your IDE.

```
> INSTRUCTOR: Set up a fresh IDE project after cloning, and
  import `amethyst-algorithms-project-1-coding-basics` with IntelliJ IDEA.
```

★ Basics of Java code: Syntax: Overview

Let's go to `/src/main/java/io/meltingscales/amethystalgorithms/DirtDropsGold.java` in our editor!

Also, for each example, please see `/src/main/java/io/meltingscales/amethystalgorithms/codingbasics/*.java`. These are more extended examples I will cover in the live video.

Java syntax can be broken up into a few distinct types. We'll cover them in the next slide.

This is a "crash course" in Java syntax. We're going to learn just enough to start modding.

- L16: A method (also function) declaration! `onBlockBreak(...){...}`
- L13: an annotation! `@EventBusSubscriber(...)`
- L40: method/function invocation.
- L14: a class declaration! `public class DirtDropsGold {...}`.
- L27: A variable declaration! `int blockX = blockPos.getX();`
- L22: An `if` statement! `if ( some_condition_is_true ) { do_some_action }`
- L22: a `boolean` expression! `blockState.getBlock() == Blocks.DIRT`
- L43: passing arguments to a constructor! `new ItemEntity((Level) level, blockX+0.5, blockY+0.5, blockZ+0.5)`
- creating a constructor! (See below...)

★ Basics of Java code: Syntax: Function/method declarations

Just like in algebra, i.e. `f(x,y) = x + (2 * y)`, a cornerstone of programming is the ability to declare functions.

I've found that most classes in Java over-focus on "Object-Oriented Programming" (OOP), but I've found that to be a waste of time. So I'm not going to focus on it!

You may hear "Method" and "Function" used interchangeably. Just know that they both mean "A named piece of code that usually does something very specific".

```java
// File: /src/main/java/io/meltingscales/amethystalgorithms/DirtDropsGold.java
public static void onBlockBreak(BlockEvent.BreakEvent event) {
    // (we drop gold and clay balls if they break a block...)
}
```

```
> INSTRUCTOR: (Open `DirtDropsGold.java`, then open `CodingBasicsFunctionsAndMethods.java`).
> INSTRUCTOR: Talk about the code.
```

★ Basics of Java code: Using constructors

Constructors are just functions that return an "object", which is a collection of data.

Again, I'm going to avoid teaching OOP. All you need to know is that it's like a function that gives you a special thing that has both data (numbers, text) and functions (things that can transform data) tied together.

Note the `new` keyword here - This is what creates the new object and invokes the constructor...

```
// File: /src/main/java/io/meltingscales/amethystalgorithms/DirtDropsGold.java
ItemStack oneGoldIngot =
    new net.minecraft.world.item.ItemStack(net.minecraft.world.item.Items.GOLD_INGOT);
```

```
> INSTRUCTOR: (Open `DirtDropsGold.java`, then open `CodingBasicsConstructors.java`).
> INSTRUCTOR: Talk about the code.
```

★ Basics of Java code: Annotations

Annotations are ways to store extra data (“metadata”, literally “data about data”) about a method (function), variable, or class.

In NeoForged, they are very powerful and mean specific things. They tell the mod framework to hook specific functions into your mod, and do things like set up “event listeners” and “keybind listeners”.

```java
// File: /src/main/java/io/meltingscales/amethystalgorithms/DirtDropsGold.java
@EventBusSubscriber(modid = AmethystAlgorithms.MODID, bus = EventBusSubscriber.Bus.GAME)
public class DirtDropsGold {

  @SubscribeEvent(receiveCanceled = true)
  public static void onBlockBreak(BlockEvent.BreakEvent event) {
    // (...)
  }
}
```

```
> INSTRUCTOR: (Open `DirtDropsGold.java`, then open `CodingBasicsAnnotations{1..3}.java`).
> INSTRUCTOR: Talk about the code.
```

Basics of Java code: Class declarations

A “class” in Java is a way to store data in what are called “instance variables” (if non-`static`) or “class variables” (if `static`).

Classes can also have functions called “methods”, as we saw above.

```java
// File: /src/main/java/io/meltingscales/amethystalgorithms/DirtDropsGold.java
public class DirtDropsGold {
    // (...)
}
```

```
> INSTRUCTOR: (Open `DirtDropsGold.java`, then open `CodingBasicsClassDeclarations.java`).
> INSTRUCTOR: Talk about the code.
```

Basics of Java code: Variable declarations

Basics of Java code: `if` statements

Basics of Java code: `boolean` expressions

Basics of Java code: method/function invocations

Basics of Java code: Variables

Basics of Java code: Mod Annotations

Basics of Java code: Syntax

```
[//]: # (Flint and steel)
[//]: # (I am steve)
[//]: # (The nether)
[//]: # (Amethyst Forest biome)
```