Why are we going to talk about constructors again?

These topics are a bit more advanced, and now that you've mastered the basics, it's time to look at these additional features.

In the next couple of videos, I'll be covering.

- Initializing final fields.
- Initializers, both static and instance.
- Record and enum constructors.

The instance initializer block

An instance initializer is a block of code declared directly in a class body.

This code gets executed when an instance of the class is created.

Instance initializers are executed, before any code in class constructors is executed.

You can have multiple initializer blocks.

They will be executed in the order they are declared.

Static Initializers

A static initializer is called the first time a class is referenced or constructed.

A class can have any number of static initialization blocks.

They can be declared anywhere in the class body.

They're called in the order they appear in the source code.

You might use this to set up some environment data or log information, that's related to the class before it can be used.

Remember, this will get executed only during the class's construction and not each instance's construction.