Hello Ballerina!!

Ballerina is a general purpose, concurrent and strongly typed programming language with both textual and graphical
syntaxes, optimized for integration.

By building around a visual programming model, Ballerina allows you to connect apps and services and create any
program via a sequence diagram. It's a unique approach that we believe will make all of us faster and more effective
at designing integration logic.

At WSO2, we create open source integration middleware. Historically, integration has been about connecting systems together.
The stuff that does this — which we build — has always been software glue; it brings a great deal of functionality to the table,
sticks the important bits together and generally stayed out of the way.

This model was built for an age when enterprises were basically collections of heavily customised,
monolithic on-premise systems that had to work together. In today's world, however,
we have so many excellent pre-built systems — especially cloud services — that we now spend most our time composing
systems out of pre-built blocks. There's always a library or a service that you can just plug in there instead of
reinventing the wheel.

In this context, integration becomes part and parcel of the actual process of building systems.
Connecting X to Y and making sure it works with Z is practically the real core of a developer's workload.
You'd think that we'd have built tools for this, but we haven't.
Our integration logic is largely done in languages meant for something else — Java, Python, and so on.
This was why we decided to create a language meant for integration, and for integration only.
Which brought us to the next problem: how would that work?