Lectura 1: **OOP Meaning – What is Object-Oriented Programming?**

While new languages are being created, existing **ones** are always being updated to make **them** better.

Although most programming languages have some similarities, each **one** has specific rules and methods which makes **it** unique. One concept that is common among many programming languages is Object Oriented Programming.

When I first came across this term, **it** was a bit confusing. **It** took me some time to really understand **it's** importance in programming. But this also doubled as an opportunity for me to learn **its** key concepts and know how important **it** is for a developer's career and being able to solve challenges.

In this article we will go over Object Oriented Programming (OOP) as a whole, without relying on a particular language. You'll learn what it is, why it's so popular as a programming paradigm, it's structure, how it works, its principles, and more.

Let's get started.

What is Object-Oriented Programming?

If you were to conduct a fast internet search on what object-oriented programming is, you'll find that OOP is defined as a programming paradigm that relies on the concept of classes and objects.

Now, for a beginner, that might be a little bit confusing – but no need to worry. I will try to explain **it** in a simplest way possible, just like the famous phrase "Explain **it** to me like I'm 5".

Here's a brief overview of what you can achieve with OOP: you can use it to structure a software program into simple, reusable code blocks (in this case usually called classes), which you then use to create individual instances of the objects.

So let's find an easier definition of object-oriented programming and learn more about **it**.