

Classes

Learn to Code with Ruby

What is a class?

- A **class** is a blueprint/template for creating objects. Ruby's built-in classes include **Array**, **Hash**, and **String**.
- Ruby's core classes cannot represent objects in our everyday life (**User**, **Car**, **Invoice**, **FinancialTransaction**).
- We define our own classes to enable the creation of **new objects** in a Ruby program. The goal of object-oriented programming (OOP) is to model real-world concepts as *objects*.
- An object is a container for data (i.e. state) with methods to process or manipulate that data.

Classes vs. Objects

- A **class** is the blueprint, the template, the schematic, the outline for an eventual object.
- A concrete **object** is called an **instance** of a class. The process of creating an object is called **instantiation**.
- In real-life terms, the **class** is the blueprint and the **object** is the house(s) we build from it.
- If we were building Ruby ourselves, we would define an **Array** class with all the methods that an array should have. Then, when we would create/instantiate an array, it would have access to all those methods because it is made from the **Array** class.