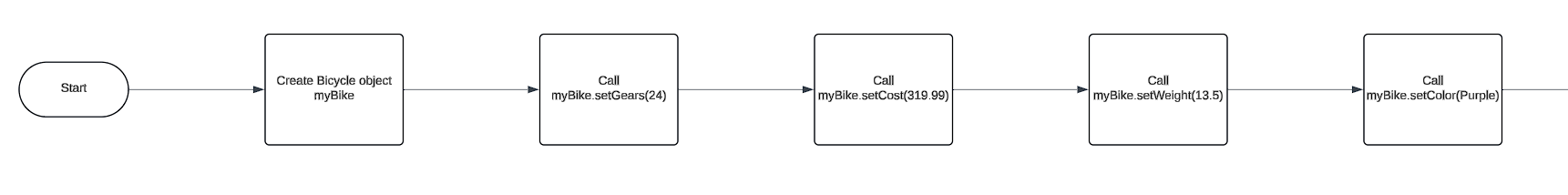
**UML Diagram**

A diagram of a bicycle

Description automatically generated

**FlowChart**



A diagram of a call center

Description automatically generated

A diagram of a diagram

Description automatically generated

Inheritance is when one class uses the properties and behaviors of another class. In this example, the Vehicle class is the parent class. TwoWheeled is a class that inherits from Vehicle, and Bicycle inherits from TwoWheeled. So, a Bicycle is a type of TwoWheeled vehicle, and a TwoWheeled vehicle is a type of Vehicle. This allows shared attributes and methods to be defined once in the base class and used by the other classes, making the code easier to manage and reuse.

Encapsulation means protecting the data inside a class. The Bicycle class has attributes like gears, cost, weight, and color. These attributes are private, so they can't be accessed directly from outside the class. Instead, the class has public methods to get and set these values. This way, the class controls how its data is accessed and changed, ensuring everything is handled properly.

Method overloading is having multiple methods with the same name but different parameters. In the Bicycle class, there are four constructors with different sets of parameters, allowing objects to be created in different ways. The outputData method is also overloaded, offering different ways to show the bicycle's details. Overloading makes the code more flexible and easier to read because the same method name can do different things based on the input.

Method chaining lets you call multiple methods in one line. The setter methods in the Bicycle class return the current object, so you can chain them together.