Homework for Iterators and Traits

Odd-Harald Lillestø Myhren

July 29, 2023

Problem 1: Understanding Iterators

Documentation for **Iterator**

- Create a vector of integers from 1 to 10.
- Use an iterator to sum up the values in the vector.
- Now, use an iterator and a closure to find all the odd numbers in the vector.
- Bonus: Implement the same logic using the `filter()` and `map()` methods available on iterators.

Problem 2: Implementing Traits

Documentation for **traits**

- Define a trait named `Drawable` with a method named `draw`.
- Implement the `Drawable` trait for a `Circle` and `Square` struct.
- For each struct, the `draw` method should return a string that describes the object. For example, "Drawing a circle..." or "Drawing a square..."
- Then, create a `Circle` and a `Square` instance and call the `draw` method from `Drawable`.

Problem 3: Trait objects

Documentation for **trait objects**

- Create a vector of `Box<dyn Drawable>` objects.
- Use `for` to loop over each element and call its `draw` method.