**WEEK 9**

**KU ID: 100061330**

**NAME: Nasser Alzaabi**

|  |
| --- |
| Q1. Rustlings exercise GitHub link: |
| ANSWER: https://github.com/NasserAlzaabi/Principles-of-Programming-Languages |
| Q2.  A. Find and explain two traits used in C2RUST? (also mention the URLs)  B. Explain where those are used, and why?  C. Explain the alternatives to using traits? |
| ANSWER:  A: One trait used is the Debug trait, it lets a type be formatted using {:?}. Which is used for debugging and lets users/developers print the state of data structures to keep track of changes and how the values of the data structure are changing. Making debugging easier.  Documentation: <https://doc.rust-lang.org/std/fmt/trait.Debug.html>  Another trait is the Clone trait, which lets you create a copy of a value. This helps in cases where ownership needs to be duplicated.  Documentation: <https://doc.rust-lang.org/std/clone/trait.Clone.html>  B:  Debug is used to enable developers to easily inspect Rust implementation of C structures which helps in making sure the translation is accurate during the migration process.  The clone trait in C2Rust is used to duplicate data structures when translating C code that needs to pass structures, which is easier through C because there is no ownership limitations, so the Clone trait enables us to mimic that C behavior.  C: There are a few alternatives. For example we can use enums which can handle different data types and let the relevant functions operate on them. We can also use function pointers as arguments, which lets us use dynamic styles without needing traits. While these alternatives could work they are not always the most efficient method and traits are valuable to the rust language. |