Console Input and Strings

Standard Library package (AKA Crate)

1. Package = Library = Crate
    1. To go from path to module or function you need to use the path separator operator
    2. `::` separated the path
2. Example There is the `io` module from `std` crate
    1. `use std::io`
3. String is a module `String`
    1. `let mut input = String::new(); //makes a new string`

Console Input

- Now to take console input
    - `io::stdin().read_line() //reads console input until '\n'`
- **Need To read in String** standard input only reads in strings
    - `io::stdin().read_line(&mut input) //passes a mutable reference to the variable input inorder to modify it and put in the user input field`
        - By default Rust passes by value like C and C++
            - i.e arguments passed into functions are passed as a copy of the value
    - read_line() returns a `Result` obj and _`.expect`_ catches error that occurs
        - `io::stdin().read_line(&mut input).expect("failed to read line");`