Rust Project Setup Guide

Step 1: Create a new Rust project

```
cargo new rust-string-example
cd rust-string-example
```

Step 2: Add the Rust program (src/main.rs)

```
// main.rs

fn print_str_slice(s: &str) {
    println!("Borrowed string slice: {}", s);
}

fn main() {
    // String: owned, growable
    let mut owned_string = String::from("Hello");
    owned_string.push_str(", Rust!"); // we can modify it
    println!("Owned String: {}", owned_string);

    // &str: borrowed, immutable string slice
    let borrowed_str: &str = "Hello, World!";
    print_str_slice(borrowed_str);

    // We can also borrow a slice from a String
    let slice_from_string: &str = &owned_string[0..5];
    print_str_slice(slice_from_string);
}
```

Step 3: Test the program

```
cargo run

Expected Output:
Owned String: Hello, Rust!
Borrowed string slice: Hello, World!
Borrowed string slice: Hello
```

Step 4: Add a README.md file

```
# Rust String vs &str Example

This is a simple Rust project that demonstrates the difference between **String** and **&str** in Rus

## About the Project
- **String** → An owned, growable, and mutable string stored on the heap.
- **&str** → A borrowed string slice that is immutable and does not own the data.

The code shows:
- How to create and modify a String.
- How to use &str as a borrowed reference.
- How to take a slice from a String and use it as &str.

## How to Run
1. Clone this repository:
   git clone https://github.com/Priyanka-Chennoju/rust-string-example.git
   cd rust-string-example

2. Run the program:
   cargo run

## Sample Output
Owned String: Hello, Rust!
```

```
Borrowed string slice: Hello, World!
Borrowed string slice: Hello

## Author
■■■ Priyanka Chennoju
GitHub | LinkedIn
```

Step 5: Push everything to GitHub

```
git init
git add .
git commit -m "Initial Rust project: String vs &str"
git branch -M main
git remote add origin https://github.com/Priyanka-Chennoju/rust-string-example.git
git push -u origin main
```