

```
let string1: &str = "Rust";  
let string2: &str = "Ceans";
```

En başta string1 ve string2 değişkenlerini başlangıç değerlerini atadık.

```
fn concatenate_strings(string1:&str, string2:&str)-> String{  
    let mut result: String = String::from(string1.clone());  
    result.push_str(string2);  
    result  
}
```

Fonksiyonumuzu ele almaya başlarsak 2 tane string dilimini değer olarak alıyor ve String ifade döndürüyormuş .En başta koda baktığımızda String tipinde result değişkenimi tanımladım. Bizim amacımız sırasıyla girilen iki kelimeyi birleştirmek olduğu için result değişkenin içine deep copy yaparak string1 ifadesinin içindeki değişkenin değeriyle atamış olduk.

Ondan sonra push_str komutu ile de result değişkene string2'yi eklemiş oluyoruz.

En sonunda result değişkenini döndürmüş oluyoruz.

```
let concatenated_string : String = concatenate_strings(string1, string2);  
println!("Result : {}", concatenated_string);
```

concatenate_string fonksiyonundan String olarak döneceği için direkt concatenated_string isimli değişkenine eşitleyebiliyoruz.

Ondan sonra da konsola yazdırıyoruz.

```
Result : RustCeans
```