



## BLOCK NONCE #10: Binding in Pattern Matching

### What?

The @ operator lets you bind a name to a value while matching it against a pattern.

### Why?

Fewer branches -> Fewer lines -> Cleaner logic.

### How?

```
if let Some(v) = value {  
    if (1..=10).contains(&v) {  
        println!("Value {v} is in range");  
    }  
}
```

```
if let Some(v @ 1..=10) = value {  
    println!("Value {v} is in range");  
}
```

You can also match: an enum!

```
match result {  
    Err(e @ RecoverableError { .. }) =>  
        println!("Failing due to recoverable error: {e:?}"),  
    Err(e @ FatalError { .. }) =>  
        println!("Fatal error! Shutting down: {e:?}"),  
    Ok(_) =>  
        println!("Success!"),  
}
```

A tuple or a pattern inside a tuple!

```
if let Some(point @ (x @ 1..i32::MAX, y @ 1..i32::MAX)) = pair {  
    println!("{} is in the first quadrant with  
            x = {} and y = {}", point, x, y);  
}
```

A struct field!

```
if let User { age: age @ 18..=30, .. } = user {  
    println!("{} is a young user with age {}", user.name, age);  
}
```

**Call For Action!** 🎉 Find one place in your code to make it cleaner using the @ operator, send a PR, and make your TL proud!