Block By Block.

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 variant!

```
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 pattern inside a tuple!

```
if let Some(point @ (x, y)) = pair {
    println!("Got point with x = {x}, y = {y}");
}
```

A struct field!

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

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

#starkcode