

# Fundamentals | Expressions

# ■ Expressions

- ◆ Rust is an expression-based language
  - Most things are evaluated and return some value
- ◆ Expression values coalesce to a single point
  - Can be used for nesting logic

## ■ Example

```
let my_num = 3;  
let is_lt_5 = if my_num < 5 {  
    true  
} else {  
    false  
};
```

```
let is_lt_5 = my_num < 5;
```

## ■ Example

```
let my_num = 3;  
let message = match my_num {  
  1 => "hello",  
  _ => "goodbye"  
};
```

# Example

```
enum Menu {  
    Burger,  
    Fries,  
    Drink,  
}
```

```
let paid = true;  
let item = Menu::Drink;  
let drink_type = "water";  
let order_placed = match item {  
    Menu::Drink => {  
        if drink_type == "water" {  
            true  
        } else {  
            false  
        }  
    }  
    _ => true,  
};
```

# ■ Recap

- ◆ Expressions allow nested logic
- ◆ *if* and *match* expressions can be nested
  - Best to not use more than two or three levels