درس ۷

Rust

enum · Option · match

main.rs

```
match opt {
  Some(v) => v,
  None => 0,
}
```

یک کورس رایگان از MyWebUniversity

برنامه‌نویسی Rust

درس ۷

enum ها، Option و match

enum ها با داده، Option به جای null، match، if let و let else



enum ها با داده

یک enum در Rust حالت‌های ممکن را فهرست می‌کند، و هر حالت می‌تواند دادهٔ خودش را داشته باشد. ...

این باعث می‌شود حالت‌های ناممکن قابل نمایش نباشند.

main.rs

```rust
enum Shape {
    Circle { r: f64 },
    Rect { w: f64, h: f64 },
    Triangle(f64, f64, f64),
}

enum Message {
    Quit,
    Move { x: i32, y: i32 },
    Write(String),
}
```

01

Option: دیگر null نیست

Rust، null ندارد. مقداری که ممکن است نباشد نوع Option دارد: یا Some با یک مقدار، یا None.

کمپایلر شما را مجبور می‌کند None را مدیریت کنید، پس crash های null pointer اصلاً نمی‌توانند ...

main.rs

```rust
fn find_student(id: u32) -> Option<String> {
    if id == 1 { Some("Mariam".to_string()) } else { None }
}

let name = find_student(7).unwrap_or("unknown".to_string());
let len = find_student(1).map(|n| n.len());   // Some(6)
```

02

match

match یک مقدار را با pattern ها مقایسه می‌کند و اولین شاخه‌ای را که جور شود اجرا می‌کند. ...

pattern ها می‌توانند داده را از enum ها بیرون بکشند، و آندرسکور با هر چیز دیگر جور می‌شود.

main.rs

```rust
fn area(s: &Shape) -> f64 {
    match s {
        Shape::Circle { r } => 3.14159 * r * r,
        Shape::Rect { w, h } => w * h,
        Shape::Triangle(a, b, c) => heron(*a, *b, *c),
    }
}
let label = match score {
    90..=100 => "A", 75..=89 => "B", _ => "C",
};
```

03

if let و let else

وقتی فقط به یک pattern اهمیت می‌دهید، if let از یک match کامل کوتاه‌تر است.

let else یک مقدار را می‌گیرد یا زود بیرون می‌شود، که مسیر اصلی را بدون تورفتگی نگه می‌دارد.

main.rs

```rust
if let Some(name) = find_student(1) {
    println!("Found {name}");
}

fn parse_age(s: &str) -> u32 {
    let Ok(age) = s.parse::<u32>() else {
        return 0;
    };
    age
}
```

04

method ها روی enum ها

enum ها، درست مثل struct ها، می‌توانند بلاک‌های impl با method داشته باشند. داخل آن‌ها روی ...

خود Option و Result فقط enum هایی از کتابخانهٔ استاندارد هستند، با method های مفید زیاد.

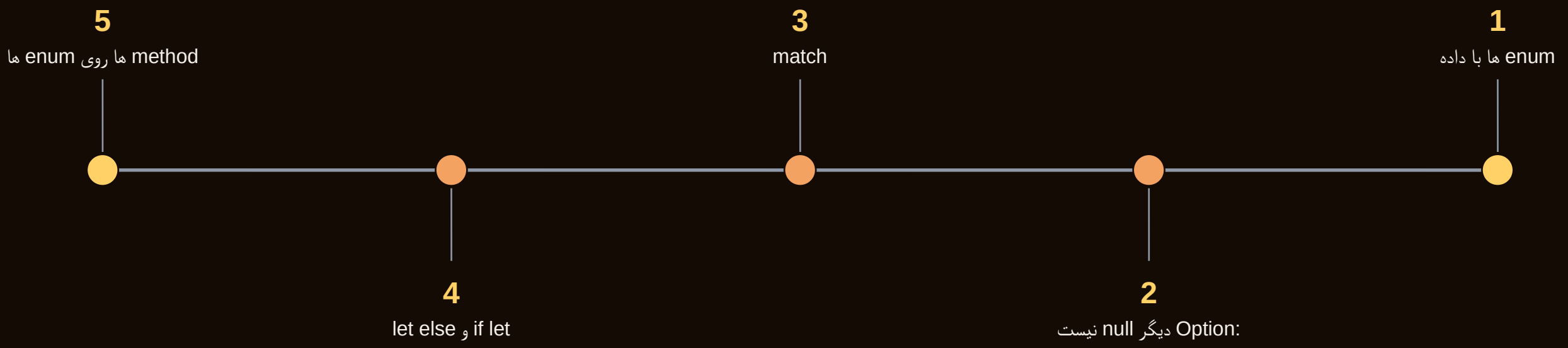
main.rs

```rust
#[derive(Debug, Clone, Copy)]
enum Light { Red, Yellow, Green }

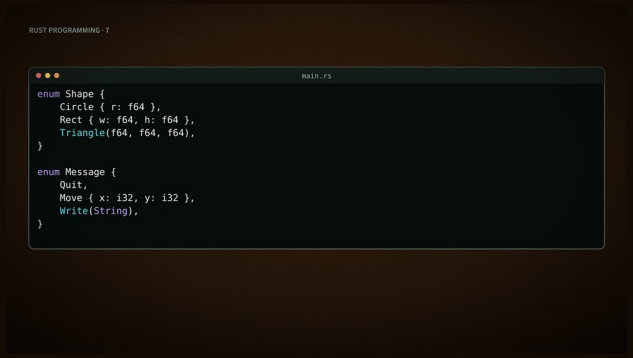
impl Light {
    fn next(self) -> Light {
        match self {
            Light::Red => Light::Green,
            Light::Green => Light::Yellow,
            Light::Yellow => Light::Red,
        }
    }
}
```

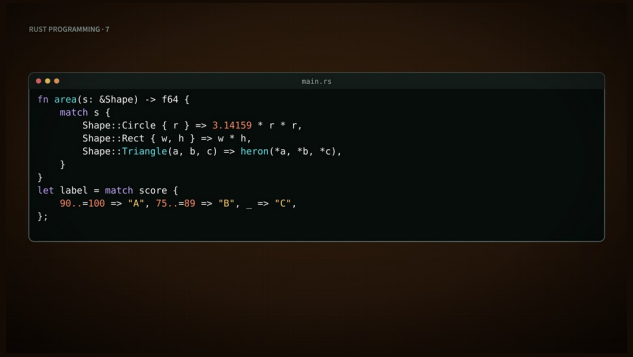
05

نقشهٔ درس


1
enum ها با داده
2
Option: دیگر null نیست
3
match
4
if let و let else
5
method ها روی enum ها

منابع و مطالعهٔ بیشتر





تشکر از تماشای شما

MyWebUniversity.com

آموزش رایگان تکنالوژی از سال ۲۰۰۰ — انگلیسی و دری

@MyWebUniversity-Dari · @MyWebUniversity