

## try\_blocks

The tracking issue for this feature is: [#31436](#)

---

The `try_blocks` feature adds support for `try` blocks. A `try` block creates a new scope one can use the `?` operator in.

```
#![feature(try_blocks)]
```

```
use std::num::ParseIntError;
```

```
let result: Result<i32, ParseIntError> = try {  
    "1".parse::<i32>()?  
    + "2".parse::<i32>()?  
    + "3".parse::<i32>()?  
};  
assert_eq!(result, Ok(6));
```

```
let result: Result<i32, ParseIntError> = try {  
    "1".parse::<i32>()?  
    + "foo".parse::<i32>()?  
    + "3".parse::<i32>()?  
};  
assert!(result.is_err());
```