

# Member access expressions

A member access expression is an expression that consists of a primary expression followed by a dot (.) and an identifier. The primary expression must have a [struct](#) type that contains a member with the identifier as its name. The type of the member access expression is the type of the member that the identifier refers to.

Example:

```
struct Point { x: usize, y: usize, }
```

```
fn foo(p: Point) -> usize { p.x }
```

[7.7 Method calls](#) [7.9 Tuple expressions](#) ð§