

The `+` type operator was used in an ambiguous context.

Erroneous code example:

```
trait Foo {}

struct Bar<'a> {
    x: &'a Foo + 'a,      // error!
    y: &'a mut Foo + 'a,  // error!
    z: fn() -> Foo + 'a,  // error!
}
```

In types, the `+` type operator has low precedence, so it is often necessary to use parentheses:

```
trait Foo {}

struct Bar<'a> {
    x: &'a (Foo + 'a),      // ok!
    y: &'a mut (Foo + 'a),  // ok!
    z: fn() -> (Foo + 'a),  // ok!
}
```

More details can be found in [RFC 438](#).