

Two associated items (like methods, associated types, associated functions, etc.) were defined with the same identifier.

Erroneous code example:

```
struct Foo(u8);

impl Foo {
    fn bar(&self) -> bool { self.0 > 5 }
    fn bar() {} // error: duplicate associated function
}

trait Baz {
    type Quux;
    fn baz(&self) -> bool;
}

impl Baz for Foo {
    type Quux = u32;

    fn baz(&self) -> bool { true }

    // error: duplicate method
    fn baz(&self) -> bool { self.0 > 5 }

    // error: duplicate associated type
    type Quux = u32;
}
```

Note, however, that items with the same name are allowed for inherent `impl` blocks that don't overlap:

```
struct Foo<T>(T);

impl Foo<u8> {
    fn bar(&self) -> bool { self.0 > 5 }
}

impl Foo<bool> {
    fn bar(&self) -> bool { self.0 }
}
```