An inherent implementation was defined for something which isn't a struct, enum, union, or trait object.

Erroneous code example:

```
impl fn(u8) { // error: no nominal type found for inherent implementation
    fn get_state(&self) -> String {
        // ...
    }
}
```

To fix this error, please implement a trait on the type or wrap it in a struct. Example:

```
// we create a trait here
trait LiveLongAndProsper {
    fn get_state(&self) -> String;
}

// and now you can implement it on fn(u8)
impl LiveLongAndProsper for fn(u8) {
    fn get_state(&self) -> String {
        "He's dead, Jim!".to_owned()
    }
}
```

Alternatively, you can create a newtype. A newtype is a wrapping tuple-struct. For example, `NewType` is a newtype over `Foo` in `struct NewType(Foo)`. Example:

```
struct TypeWrapper(fn(u8));

impl TypeWrapper {
    fn get_state(&self) -> String {
        "Fascinating!".to_owned()
    }
}
```