A reference to a local variable was returned.

Erroneous code example:

```
fn get_dangling_reference() -> &'static i32 {
    let x = 0;
    &x
}
```

```
use std::slice::Iter;
fn get_dangling_iterator<'a>() -> Iter<'a, i32> {
    let v = vec![1, 2, 3];
    v.iter()
}
```

Local variables, function parameters and temporaries are all dropped before the end of the function body. So a reference to them cannot be returned.

Consider returning an owned value instead:

```
use std::vec::IntoIter;

fn get_integer() -> i32 {
    let x = 0;
    x
}

fn get_owned_iterator() -> IntoIter<i32> {
    let v = vec![1, 2, 3];
    v.into_iter()
}
```