

# Scope

```
#!/usr/bin/perl
```

```
use strict;  
use warnings;
```

```
my $x = 100;  
my $y = 20;
```

```
if ($x > $y) {  
    my $z = 10;  
    $x = 30;  
    print "x (inside if block): $x\n";  
    print "y (inside if block): $y\n";  
    print "z (inside if block): $z\n";  
}
```

```
print "x (outside if block): $x\n";  
print "y (outside if block): $y\n";  
print "z (outside if block): $z\n";
```

Global symbol "\$z" requires explicit package name at ./scope.pl line 19.

Execution of ./scope.pl aborted due to compilation errors.

# Blocks

That's because `$z` was declared inside the if block, so it's only accessible inside that block.

Any time we see `{ }`, we're creating a block.

Blocks are like boxes that have one way mirrors – you can see outside the box from inside, but not inside the box from the outside.

To fix that error, we need to declare `$z` outside the if block.

# Blocks

Variables declared inside of a block using “my” only exist inside the block – once the block is finished, they will be destroyed.

```
#!/usr/bin/perl
```

```
use strict;  
use warnings;
```

```
my $x = 100;  
my $y = 20;  
my $z = 5;
```

```
if ($x > 10) {  
    my $z = 10;  
    $x = 30;  
    print "x (inside if block): $x\n";  
    print "y (inside if block): $y\n";  
    print "z (inside if block): $z\n";  
}
```

```
print "x (outside if block): $x\n";  
print "y (outside if block): $y\n";  
print "z (outside if block): $z\n";
```

Output:

```
$x (inside of block):30  
$z (inside of block):10  
$x (outside if block): 30  
$z (outside if block):5
```

## Scope

Does the program give the expected behavior?

By declaring “`my $z = 10;`” inside the if block, we’re creating a new variable called `$z` only accessible within the block.

This new variable will not modify the outside variable!

Note that we can create a new `$z` variable inside the block with no problems – if we do it outside, we’ll get a warning.

# Scope

- If we remove “my” from that line, the modification to \$z will show outside the block.

```
#!/usr/bin/perl
```

```
use strict;  
use warnings;
```

```
my $x = 100;  
my $z = 5;
```

```
if ($x > 10) {  
    $z = 10;  
    $x = 30;  
    print "x (inside if block): $x\n";  
    print "z (inside if block): $z\n";  
}
```

```
print "x (outside if block): $x\n";  
print "z (outside if block): $z\n";
```

Output:

```
$x (inside if block): 30  
$z (inside if block): 10  
$x (outside if block): 30  
$z (outside if block): 10
```