

Unit 3—Lesson 4:

Constant and Variable Scope

Scope

Global scope—Defined outside of a function

Local scope—Defined within braces (`{ }`)

```
var globalVariable = true

if globalVariable {
  let localVariable = 7
}
```

Scope

```
var age = 55
```

```
func printMyAge() {  
    print("My age: \(age)")  
}
```

```
print(age)  
printMyAge()
```

55

My age: 55

Scope

```
func printBottleCount() {  
    let bottleCount = 99  
    print(bottleCount)  
}
```

```
printBottleCount()  
print(bottleCount)
```



Use of unresolved identifier 'bottleCount'

Scope

```
func printTenNames() {  
    var name = "Richard"  
    for index in 1...10 {  
        print("\(index): \(name)")  
    }  
    print(index)  
    print(name)  
}
```

```
printTenNames()
```



Use of unresolved identifier 'index'

Variable shadowing

```
let points = 100

for index in 1...3 {
    let points = 200
    print("Loop \(index): \(points+index)")
}
print(points)
```

```
Loop 1: 201
Loop 2: 202
Loop 3: 203
100
```

Variable shadowing

```
var name: String? = "Robert"

if let name = name {
    print("My name is \(name)")
}
```

Variable shadowing

```
func exclaim(name: String?) {  
    if let name = name {  
        print("Exclaim function was passed: \(name)")  
    }  
}
```

```
func exclaim(name: String?) {  
    guard let name = name else { return }  
    print("Exclaim function was passed: \(name)")  
}
```


Shadowing and initializers

```
struct Person {  
    var name: String  
    var age: Int  
}  
  
let todd = Person(name: "Todd", age: 50)  
print(todd.name)  
print(todd.age)
```

Todd

50

Shadowing and initializers

```
struct Person {  
    var name: String  
    var age: Int  
  
    init(name: String, age: Int) {  
        self.name = name  
        self.age = age  
    }  
}
```

Unit 3—Lesson 4

Lab: Constant and Variable Scope



Open and complete the exercises in Lab – Scope.playground

