Level 2-2

# Underneath the Tracks

**Functions** 



#### The Greeting Program

all the control of the control of the first of the first of the control of the co

Let's write a program that **outputs a greeting message**. The message will be different depending on which hour of the day we run our program.

Market & Sharehall, in the Control of the State of the Control of



# Grabbing Current Time

CONTRACTOR OF THE STATE OF THE

We'll start by grabbing the current hour of the day.

```
src/hello/main.go
package main
                  Import package from standard library.
import (
  "fmt"
  "time"
                      Assign return value to a new
                      variable using type inference.
func main()
  hourOfDay := time.Now().Hour()
```

# The getGreeting() Function Signature

Named functions in Go must have **a name**, followed by **any arguments** they expect, and end with the **data type** they return (if any). This is commonly referred to as a **function signature**.

```
src/hello/main.go
func main() {
  hourOfDay := time.Now().Hour()
  greeting := getGreeting(hourOfDay)
    How do we find out which
    data type this is?
func getGreeting(hour [
```

Marie Company Comment of the Comment

Calling the new function passing the current hour of the day as its single argument

Expected return type for this function

Markey Charles Contract to Contract the Contract to Co

#### The Return Data Type

With the time of the state of t

One way to find out the **data type returned by a function** from the Go standard library is by referring to the official documentation.

```
src/hello/main.go
                                                                 http://go.codeschool.com/go-time-package
                                                                   https://golang.org/pkg/time/
                                                                  nunc (t mine) ronnattiayout string) string
                                                                  func (t *Time) GobDecode(data []byte) error
func main() {
                                                                  func (t Time) GobEncode() ([]byte, error)
   hourOfDay := time.Now().Hour()
                                                                  func (t Time) Hour() int
                                                                  func (t Time) ISOWeek() (year, week int)
   greeting := getGreeting(hourOfDay)
                                                                  func (t Time) In(loc *Location) Time
                                                            According to the docs for the time package,
                                                            Hour() returns a value of type int.
func getGreeting(hour int) string {
                                                     The complete function signature
```

Markey & Standard and Amelia and Deline the Markey Company of the land of the Markey Company of the Company of

## Returning From getGreeting()

We'll return a different greeting message based on the hour of the day passed as argument.

```
src/hello/main.go
func getGreeting(hour int) string {
  if hour < 12 {
    return "Good Morning"
                                         We can nest ifs
  } else if hour < 18 {</pre>
                                         inside else clauses.
    return "Good Afternoon"
  } else {
    return "Good Evening"
```

The state of the s

## Returning From getGreeting()

Marie Carlo Ca

We run our program and see a different output depending on the current time of day.

```
src/hello/main.go
                                                             go run main.go
                                                                 Good Morning
func main() {
                                                               run main.go
  hourOfDay := time.Now().Hour()
  greeting := getGreeting(hourOfDay)
                                                                Good Afternoon
  fmt.Println(greeting)
                                                               run main.go
func getGreeting(hour int) string {
```