**Variables**

// Declare three variables that are initialized to their zero value and three

// declared with a literal value. Declare variables of type string, int and

// bool. Display the values of those variables.

//

// Declare a new variable of type float32 and initialize the variable by

// converting the literal value of Pi (3.14).

package main

import "fmt"

func main() {

// Declare variables that are set to their zero value.

var age int

var name string

var legal bool

// Display the value of those variables.

fmt.Println(age)

fmt.Println(name)

fmt.Println(legal)

// Declare variables and initialize.

// Using the short variable declaration operator.

month := 10

dayOfWeek := "Tuesday"

happy := true

// Display the value of those variables.

fmt.Println(month)

fmt.Println(dayOfWeek)

fmt.Println(happy)

// Perform a type conversion.

pi := float32(3.14)

// Display the value of that variable.

fmt.Printf("%T [%v]\n", pi, pi)

}

**Struct Types**

// Declare a struct type to maintain information about a user (name, email and age).

// Create a value of this type, initialize with values and display each field.

//

// Declare and initialize an anonymous struct type with the same three fields. Display the value.

package main

import "fmt"

// user represents a user in the system.

type user struct {

name string

email string

age int

}

func main() {

// Declare variable of type user and init using a struct literal.

bill := user{

name: "Bill",

email: "bill@ardanlabs.com",

age: 45,

}

// Display the field values.

fmt.Println("Name", bill.name)

fmt.Println("Email", bill.email)

fmt.Println("Age", bill.age)

// Declare a variable using an anonymous struct.

ed := struct {

name string

email string

age int

}{

name: "Ed",

email: "ed@ardanlabs.com",

age: 46,

}

// Display the field values.

fmt.Println("Name", ed.name)

fmt.Println("Email", ed.email)

fmt.Println("Age", ed.age)

}

**Pointers**

Ex1

// Declare and initialize a variable of type int with the value of 20. Display

// the \_address of\_ and \_value of\_ the variable.

//

// Declare and initialize a pointer variable of type int that points to the last

// variable you just created. Display the \_address of\_ , \_value of\_ and the

// \_value that the pointer points to\_.

package main

import "fmt"

func main() {

// Declare an integer variable with the value of 20.

value := 20

// Display the address of and value of the variable.

fmt.Println("Address Of:", &value, "Value Of:", value)

// Declare a pointer variable of type int. Assign the

// address of the integer variable above.

p := &value

// Display the address of, value of and the value the pointer

// points to.

fmt.Println("Address Of:", &p, "Value Of:", p, "Points To:", \*p)

}

Ex2

// Declare a struct type and create a value of this type. Declare a function

// that can change the value of some field in this struct type. Display the

// value before and after the call to your function.

package main

import "fmt"

// user represents a user in the system.

type user struct {

name string

email string

accessLevel int

}

func main() {

// Create a variable of type user and initialize each field.

bill := user{

name: "Bill",

email: "bill@ardanlabs.com",

accessLevel: 1,

}

// Display the value of the accessLevel field.

fmt.Println("access:", bill.accessLevel)

// Share the bill variable with the accessLevel function

// along with a value to update the accessLevel field with.

accessLevel(&bill, 10)

// Display the value of the accessLevel field again.

fmt.Println("access:", bill.accessLevel)

}

// accessLevel changes the value of the users access level.

func accessLevel(u \*user, accessLevel int) {

// Set of value of the accessLevel field to the value

// that is passed in.

u.accessLevel = accessLevel

}

**Constants**

// Declare an untyped and typed constant and display their values.

//

// Multiply two literal constants into a typed variable and display the value.

package main

import "fmt"

const (

// server is the IP address for connecting.

server = "124.53.24.123"

// port is the port to make that connection.

port int16 = 9000

)

func main() {

// Display the server information.

fmt.Println(server)

fmt.Println(port)

// Calculate the number of minutes in 5320 seconds.

minutes := 5320 / 60.0

fmt.Println(minutes)

}