Day 20 - Go (Golang Basics) - Conditional Structures

If Statements

An `if` statement is used to execute a block of code if a condition is true.

Syntax:

```
if <condition> {
    // executes when condition is true
}
```

Example:

```go
package main

import "fmt"

func main() {
    var fruit string = "apple"

if fruit == "apple" {
    fmt.Println("Fruit is apple")
}

}
```

If-Else Statement

An `if-else` block provides two paths: one for true, one for false.

Syntax:

```
if <condition> {
    // executes when condition is true
} else {
    // executes when condition is false
}
```

Example:

```go
package main

import "fmt"

func main() {
    var fruit string = "apple"

if fruit == "apple" {
    fmt.Println("Fruit is apple")
} else {
    fmt.Println("Fruit is not apple")
}

}
```

If-Else If-Else Statement

Used to check multiple conditions.

Syntax:

```
if <condition1> {
    // executes if condition1 is true
} else if <condition2> {
    // executes if condition2 is true
} else {
    // executes if neither is true
}
```

Example:

```go
package main

import "fmt"

func main() {
    var fruit string = "apple"

if fruit == "banana" {
    fmt.Println("Fruit is banana")
} else if fruit == "apple" {
    fmt.Println("Fruit is apple")
} else {
    fmt.Println("Fruit is neither banana nor apple")
}

}
```

Switch Statement

A switch statement allows you to perform different actions based on the value of a variable or expression.

Syntax:

```
switch variable {
case value1:
    // code block
case value2:
    // code block
default:
    // default block
}
```

Example:

```go
package main

import "fmt"

func main() {
    fruit := "apple"

```

```
switch fruit {
case "banana":
    fmt.Println("Fruit is banana")
case "apple":
    fmt.Println("Fruit is apple")
default:
    fmt.Println("Unknown fruit")
}

}
```

Fallthrough

The `fallthrough` keyword is used to continue executing the next case even if it doesn't match.

Example:

```
package main

import "fmt"

func main() {
    number := 2

switch number {
case 1:
    fmt.Println("One")
case 2:
    fmt.Println("Two")
    fallthrough
case 3:
    fmt.Println("Three")
default:
    fmt.Println("Unknown")
}

}
```

Switch With Conditions

You can use conditions in switch statements without providing a variable.

Example:

```
package main

import "fmt"

func main() {
    age := 18

switch {
case age < 13:
    fmt.Println("Child")
case age >= 13 && age < 20:
    fmt.Println("Teenager")
case age >= 20:
    fmt.Println("Adult")
```

```
}

}
```