2 - Your First API Server

Building an API Web Server

src/first-api-server/main.go

```
package main

import (
    "fmt"
    "net/http"
)

func main() {
    err := http.ListenAndServe("localhost:8080", nil)
    if err != nil {
        fmt.Println(err)
    }
}
```

Creating the Default Route

Creating the Default Hadler

src/first-api-server/main.go

```
package main

import (
    "fmt"
    "net/http"
    "os"
)

func rootHandler(w http.ResponseWriter, r *http.Request) {
    w.WriteHeader(http.StatusOK)
    w.Write([]byte("Running API v1\n"))
}

func main() {
    http.HandleFunc("/", rootHandler)
    err := http.ListenAndServe("localhost:8080", nil)
    if err != nil {
        fmt.Println(err)
        os.Exit(1)
    }
}
```

Custom Error Message

src/first-api-server/main.go

```
package main

import (
    "fmt"
    "net/http"
    "os"
)

func rootHandler(w http.ResponseWriter, r *http.Request) {
    if r.URL.Path != "/" {
        w.WriteHeader(http.StatusNotFound)
        w.Write([]byte("Asset not found\n"))
        return
    }
    w.WriteHeader(http.StatusOK)
    w.Write([]byte("Running API v1\n"))
}

func main() {
    http.HandleFunc("/", rootHandler)
    err := http.ListenAndServe("localhost:8080", nil)
    if err != nil {
        fmt.Println(err)
        os.Exit(1)
    }
}
```

testcase 1: (Normal)

```
$ curl localhost:8080
```

testcase 2: (Error)

```
$ curl localhost:8080/something
```