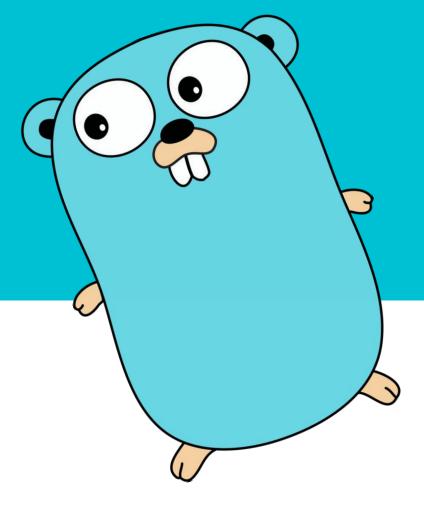# Golang
HTTP
REST / JSON

Laurent Guérin
Version 1.0 - 2019 October

Conversion : structure ← → JSON

```go
// Generated by Telosys Tools Generator ( version 3.1.0 )
package rest

import (
    "net/http"
    "encoding/json"
)


func WriteJSON(w http.ResponseWriter, data interface{}) {
    jsonData, err := json.Marshal(data)
    if err != nil {
        http.Error(w, err.Error(), http.StatusInternalServerError)
        return
    }
    w.Header().Set("Content-Type", "application/json")
    w.Write(jsonData)
}


func ReadJSON(data interface{}, r *http.Request) error {
    defer r.Body.Close()
    return json.NewDecoder(r.Body).Decode(data)
}
```

Any structure

HTTP Status

```go
// GET
func ReplyFound(w http.ResponseWriter) {
    w.WriteHeader(http.StatusOK)
}
func ReplyNotFound(w http.ResponseWriter) {
    w.WriteHeader(http.StatusNotFound)
}

// CREATE
func ReplyCreated(w http.ResponseWriter) {
    // StatusCreated = 201 // RFC 7231, 6.3.2
    // The 201 (Created) status code indicates that the request has been
    // fulfilled and has resulted in one or more new resources being created.
    w.WriteHeader(http.StatusCreated)
}

func ReplyNotCreated(w http.ResponseWriter) {
    // StatusConflict = 409 // RFC 7231, 6.5.8
    // The 409 (Conflict) status code indicates that the request could not
    // be completed due to a conflict with the current state of the target resource.
    w.WriteHeader(http.StatusConflict)
}
```

xxx