

Echo HTTP 方法

1. 概要

DaboryGo による Echo 方法の概要を、JSON 形式の c.String() と c.JSONBlob() を含む HTTP 方法の json 形式の

2. c.String(statusCode int, response string) 方法

概要

- c.String() は、statusCode と response を返す。
- statusCode は HTTP 方法の response を返す。

例

```
var c echo.Context
return c.String(607, "")
```

例

- 607 は HTTP 方法。
 - "" は response 。
 - response.status は 607 。
-

3. c.JSONBlob(statusCode int, jsonData []byte) 方法

概要

- c.JSONBlob() は JSON 形式の jsonData を返す。
- statusCode は HTTP 方法の jsonData を返す。

□ □ □

```
ret, err := json.Marshal(vRet)
if err != nil {
    return c.String(500, "Failed to encode JSON: "+err.Error())
}
return c.JSONBlob(http.StatusOK, ret)
```

□ □ □

1. `json.Marshal(vRet)` □ `ZbaksanSorderEyetestRes` □ □ □ □ JSON □ □ □ □.
2. □ □ □ JS(`c.JSONBlob(http.StatusOK, ret)`) □ □ □ □ □ □ □ □.
3. □ `response.json()` □ `response.data` □ **JSON** □ □ □ □.

4. `c.String()` vs `c.JSONBlob()` □ □

□ □	<code>c.String()</code>	<code>c.JSONBlob()</code>
□ □ □	<code>text(String)</code>	JSON (□ □ □ □)
□ □ □	□ □ □ □, □ □ □ □	JSON □ □ □ □
□ □ □	<code>c.String(607, "")</code>	<code>c.JSONBlob(http.StatusOK, ret)</code>
□ □ □ □ □	<code>response.text()</code>	<code>response.json()</code> □ <code>response.data</code>

5. □ □

□ `c.String(607, "")`

- HTTP □ `607` □ □ □ □ □ □ □ □ □ □.
- □ `response.status` □ □ □ □ □ □ □ □.

□ `c.JSONBlob(http.StatusOK, ret)`

- JSON □ □ □ □ □ □ □ □.
- `json.Marshal()` □ □ □ □ JSON `response.json()` □ □ □ □.

