LATIHAN PERTEMUAN 8

PRAKTIKUM PEMOGRAMAN BERBASIS WEB

Untuk Memenuhi Praktikum Pemograman Berbasis Web

Oleh:

Nama	: Siti Ghumaisa
NPM	: 4522210131
Kelas	: A
Semester	: 4 (Genap)

Dosen :

ADI WAHYU PRIBADI ,S.SI.,M.KOM

S1-Teknik Informatika

Fakultas Teknik Universitas Pancasila

2023/2024

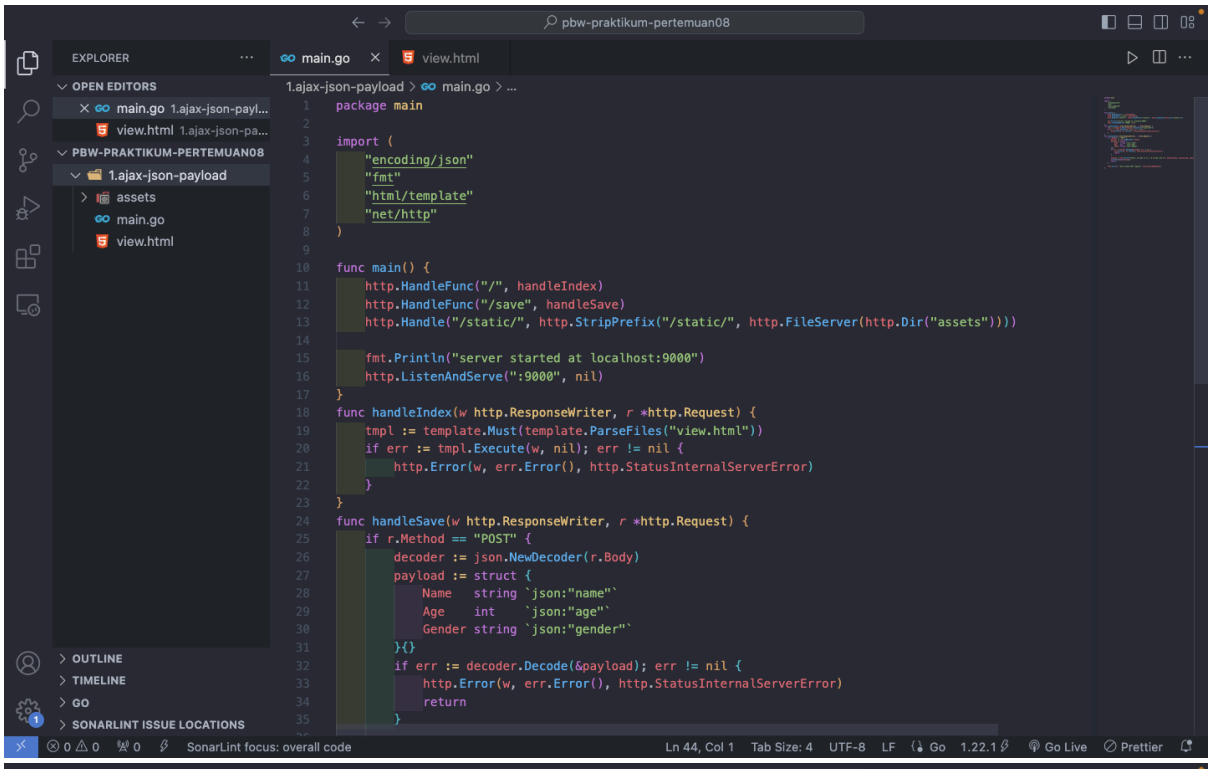
Link git-hub https://github.com/Ghumaisa/PBW

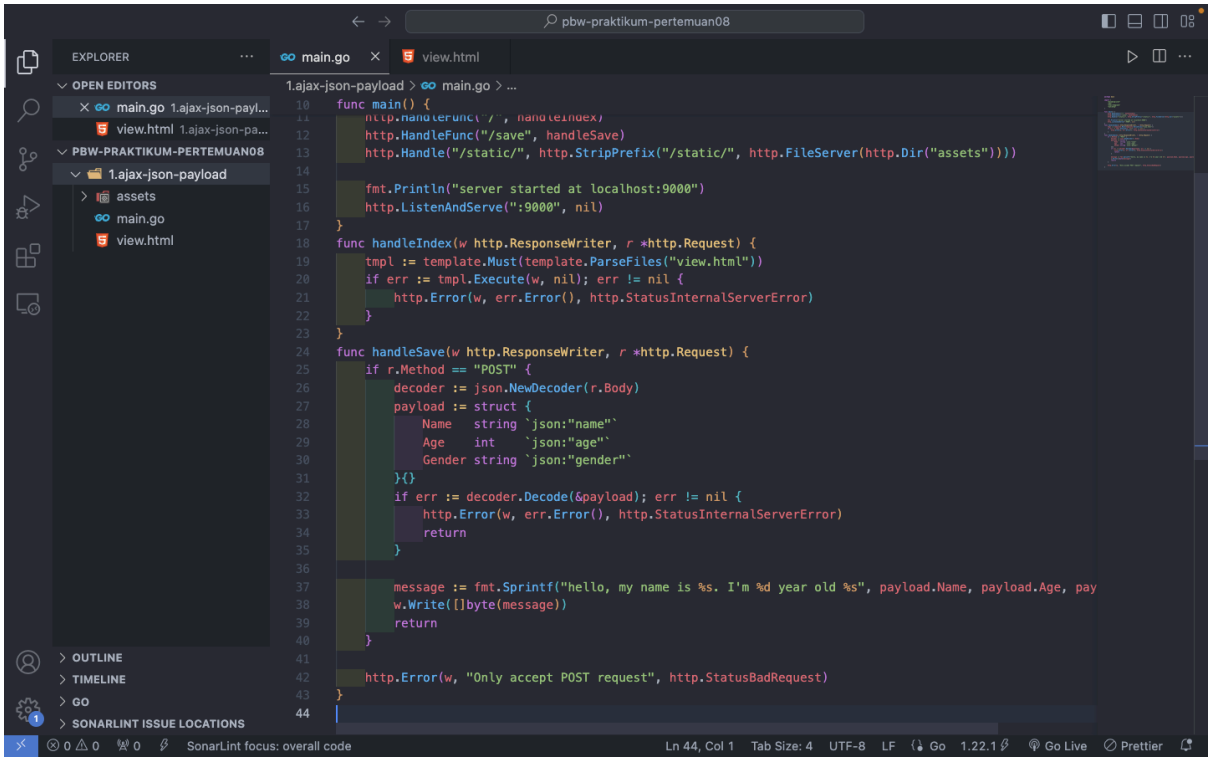
MATERI PRAKTIKUM :

1. AJAX JSON Payload

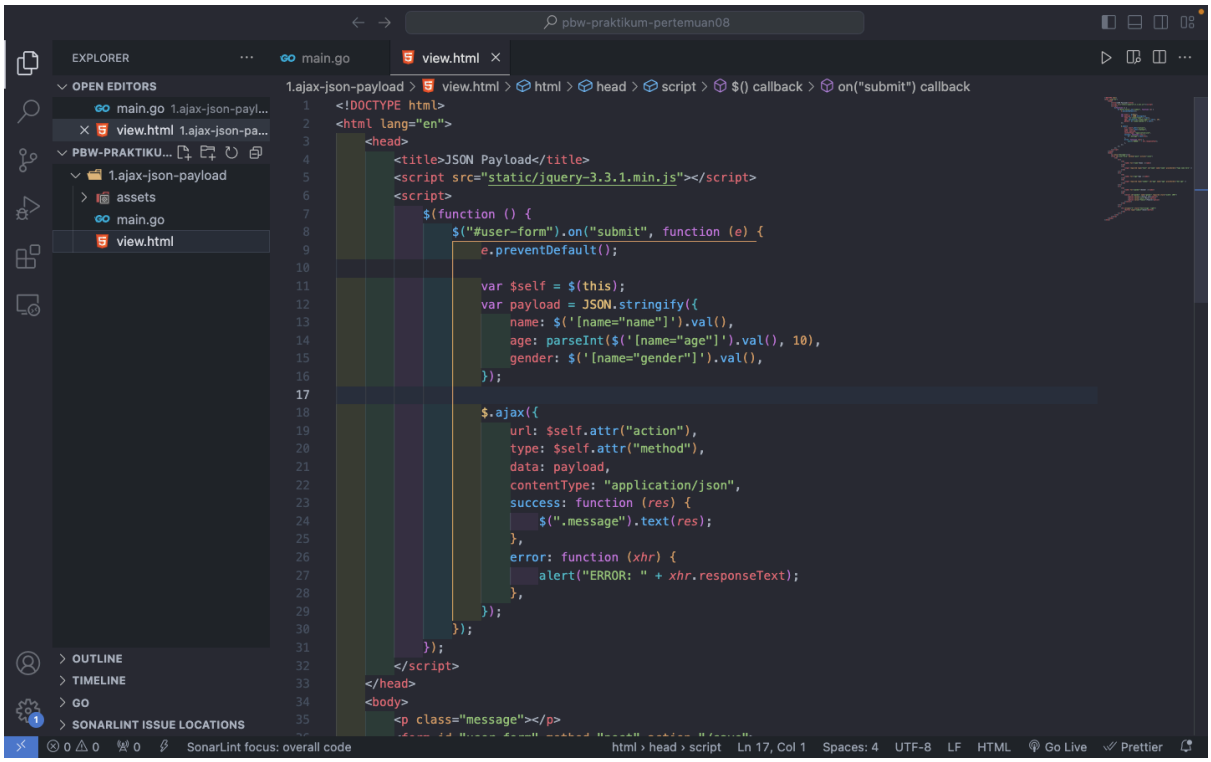
   Screenshot Source Code

   main.go

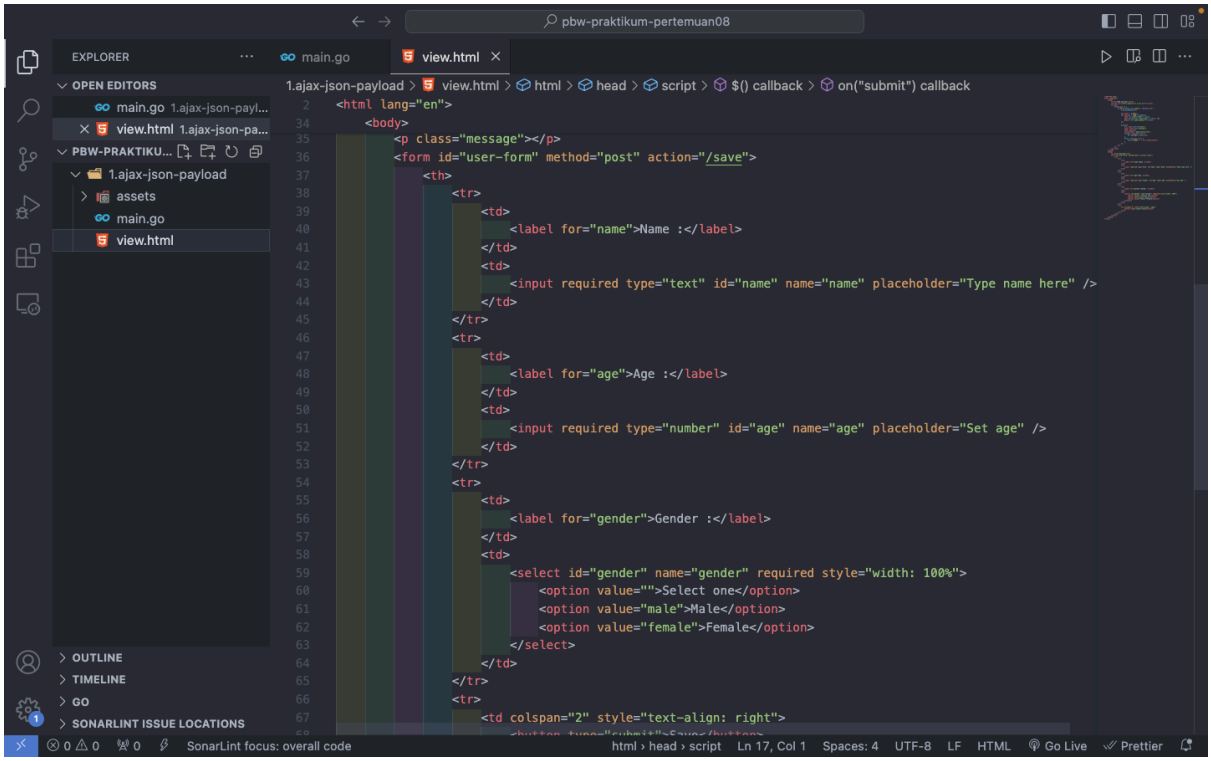




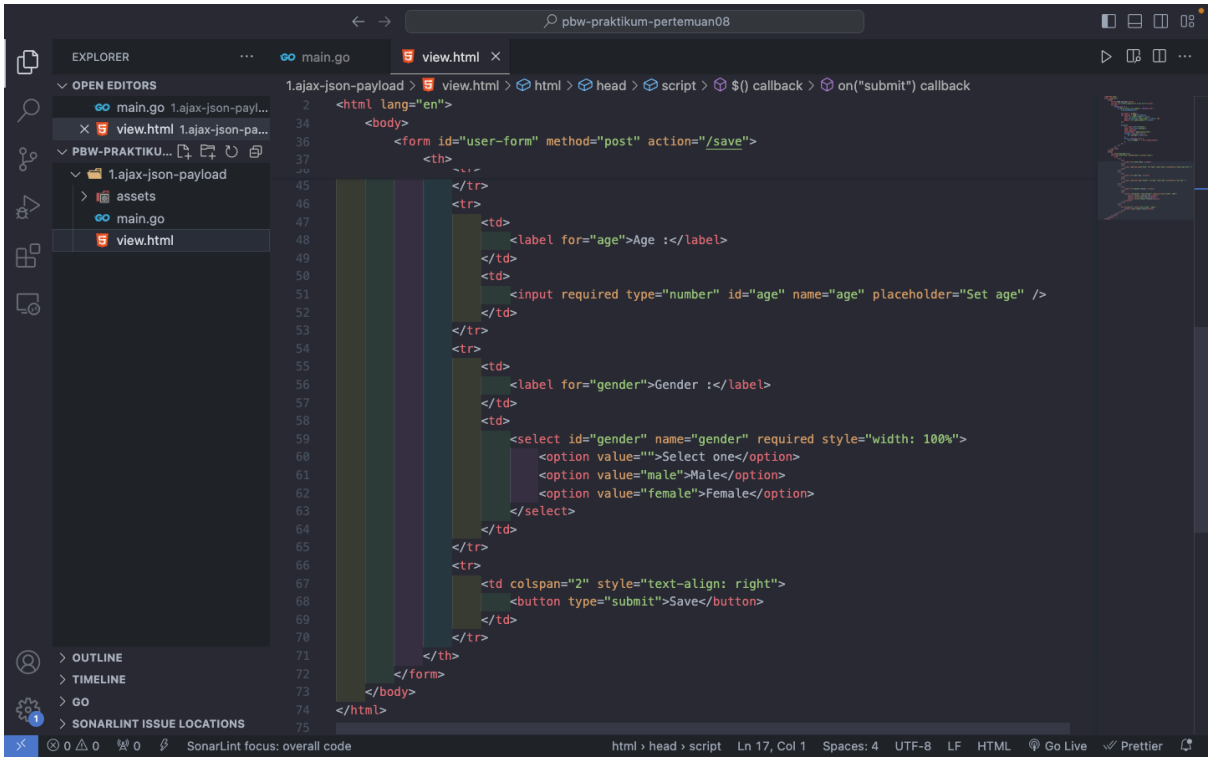
view.html



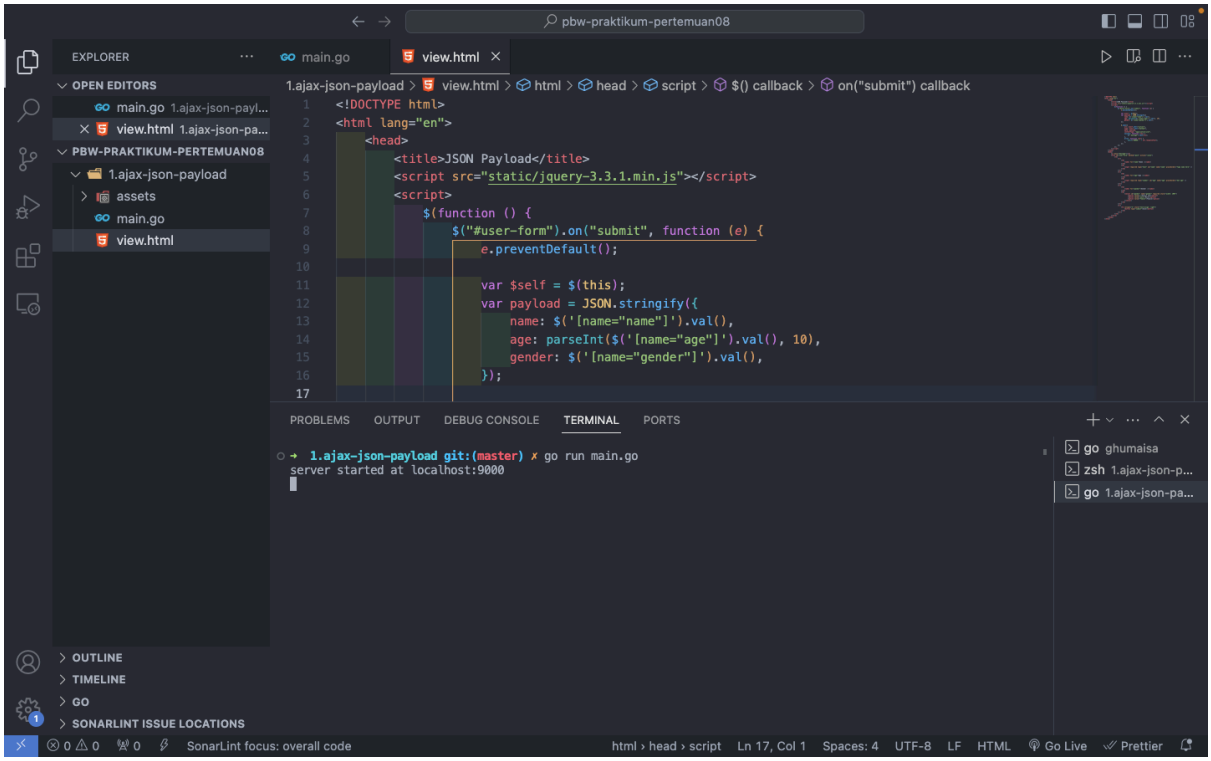
```
<!DOCTYPE html>
<html lang="en">
    <head>
        <title>JSON Payload</title>
        <script src="static/jquery-3.3.1.min.js"></script>
        <script>
            $(function () {
                $("#user-form").on("submit", function (e) {
                    e.preventDefault();

                    var $self = $(this);
                    var payload = JSON.stringify({
                        name: $('[name="name"]').val(),
                        age: parseInt($('[name="age"]').val(), 10),
                        gender: $('[name="gender"]').val(),
                    });

                    $.ajax({
                        url: $self.attr("action"),
                        type: $self.attr("method"),
                        data: payload,
                        contentType: "application/json",
                        success: function (res) {
                            $(".message").text(res);
                        },
                        error: function (xhr) {
                            alert("ERROR: " + xhr.responseText);
                        },
                    });
                });
            });
        </script>
    </head>
    <body>
        <p class="message"></p>
```



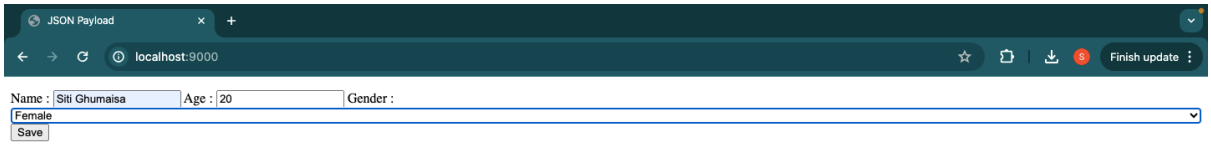
```
<html lang="en">
    <body>
        <p class="message"></p>
        <form id="user-form" method="post" action="/save">
            <th>
                <tr>
                    <td>
                        <label for="name">Name :</label>
                    </td>
                    <td>
                        <input required type="text" id="name" name="name" placeholder="Type name here" />
                    </td>
                </tr>
                <tr>
                    <td>
                        <label for="age">Age :</label>
                    </td>
                    <td>
                        <input required type="number" id="age" name="age" placeholder="Set age" />
                    </td>
                </tr>
                <tr>
                    <td>
                        <label for="gender">Gender :</label>
                    </td>
                    <td>
                        <select id="gender" name="gender" required style="width: 100%">
                            <option value="">Select one</option>
                            <option value="male">Male</option>
                            <option value="female">Female</option>
                        </select>
                    </td>
                </tr>
                <tr>
                    <td colspan="2" style="text-align: right">
```


pbw-praktikum-pertemuan08
EXPLORER
main.go
view.html
OPEN EDITORS
main.go 1.ajax-json-payl...
view.html 1.ajax-json-pa...
PBW-PRAKTIKU...
1.ajax-json-payload
assets
main.go
view.html
1.ajax-json-payload > view.html > html > head > script > $() callback > on("submit") callback
<html lang="en">
<body>
<form id="user-form" method="post" action="/save">
<th>
</tr>
<tr>
<td>
<label for="age">Age :</label>
</td>
<td>
<input required type="number" id="age" name="age" placeholder="Set age" />
</td>
</tr>
<tr>
<td>
<label for="gender">Gender :</label>
</td>
<td>
<select id="gender" name="gender" required style="width: 100%">
<option value="">Select one</option>
<option value="male">Male</option>
<option value="female">Female</option>
</select>
</td>
</tr>
<tr>
<td colspan="2" style="text-align: right">
<button type="submit">Save</button>
</td>
</tr>
</th>
</form>
</body>
</html>
OUTLINE
TIMELINE
GO
SONARLINT ISSUE LOCATIONS
SonarLint focus: overall code
html › head › script
Ln 17, Col 1
Spaces: 4
UTF-8
LF
HTML
Go Live
Prettier

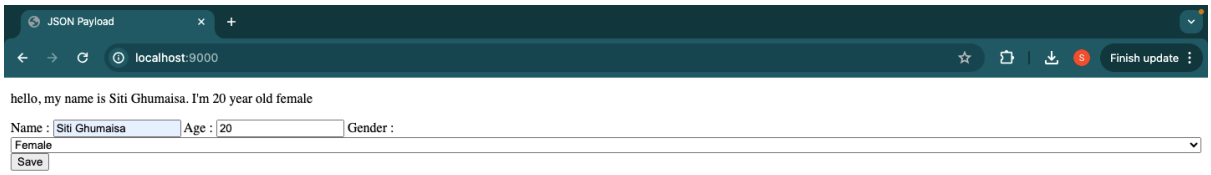
Output

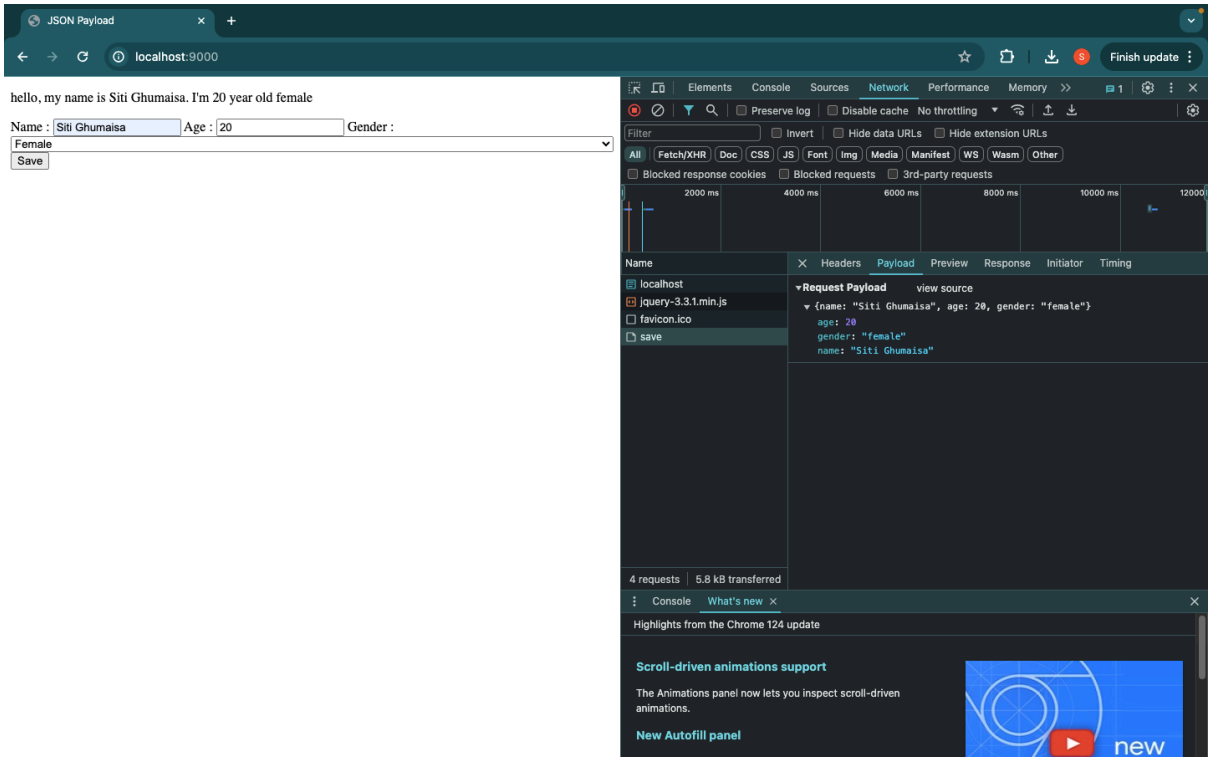

pbw-praktikum-pertemuan08
EXPLORER
main.go
view.html
OPEN EDITORS
main.go 1.ajax-json-payl...
view.html 1.ajax-json-pa...
PBW-PRAKTIKUM-PERTEMUAN08
1.ajax-json-payload
assets
main.go
view.html
1.ajax-json-payload > view.html > html > head > script > $() callback > on("submit") callback
<!DOCTYPE html>
<html lang="en">
<head>
<title>JSON Payload</title>
<script src="static/jquery-3.3.1.min.js"></script>
<script>
$(function () {
$("#user-form").on("submit", function (e) {
e.preventDefault();
var $self = $(this);
var payload = JSON.stringify({
name: $('[name="name"]').val(),
age: parseInt($('[name="age"]').val(), 10),
gender: $('[name="gender"]').val(),
});
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
1.ajax-json-payload git:(master) ✗ go run main.go
server started at localhost:9000
go ghumaisa
zsh 1.ajax-json-p...
go 1.ajax-json-pa...
OUTLINE
TIMELINE
GO
SONARLINT ISSUE LOCATIONS
SonarLint focus: overall code
html › head › script
Ln 17, Col 1
Spaces: 4
UTF-8
LF
HTML
Go Live
Prettier

Memasukkan “Nama”, “Age”, dan “Gender”



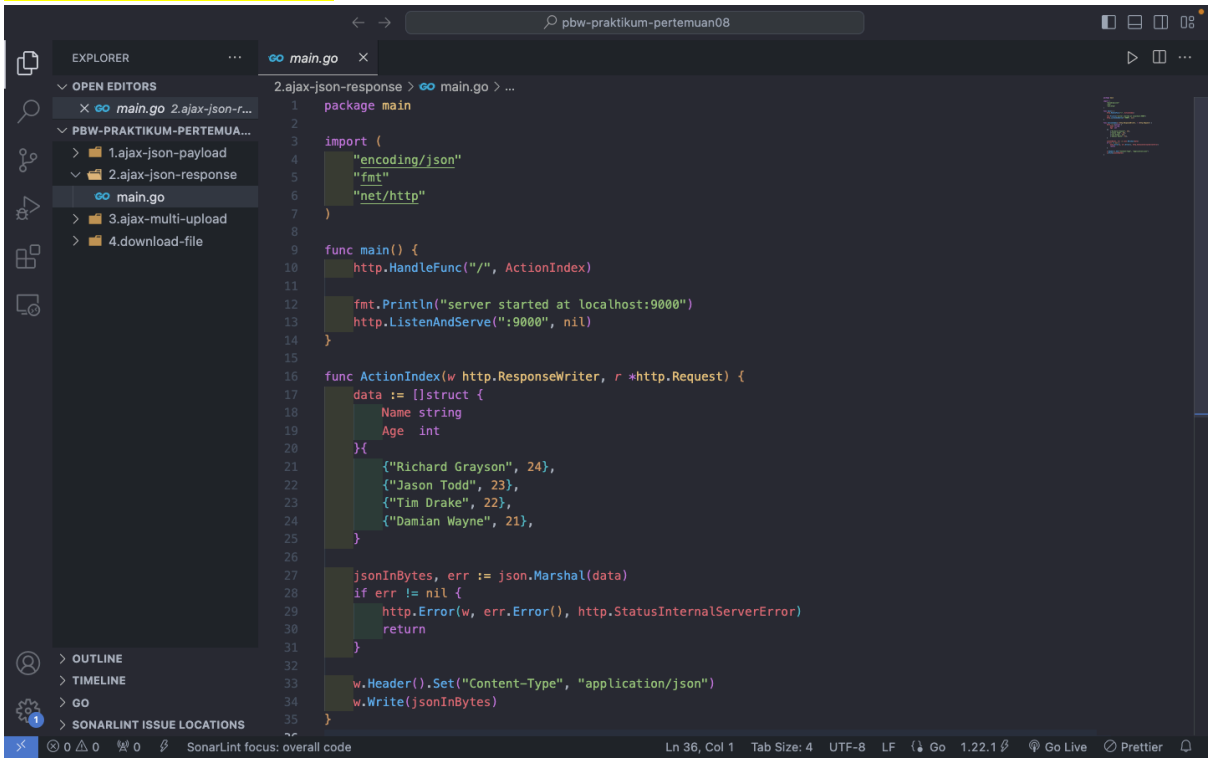
Setelah di klik save



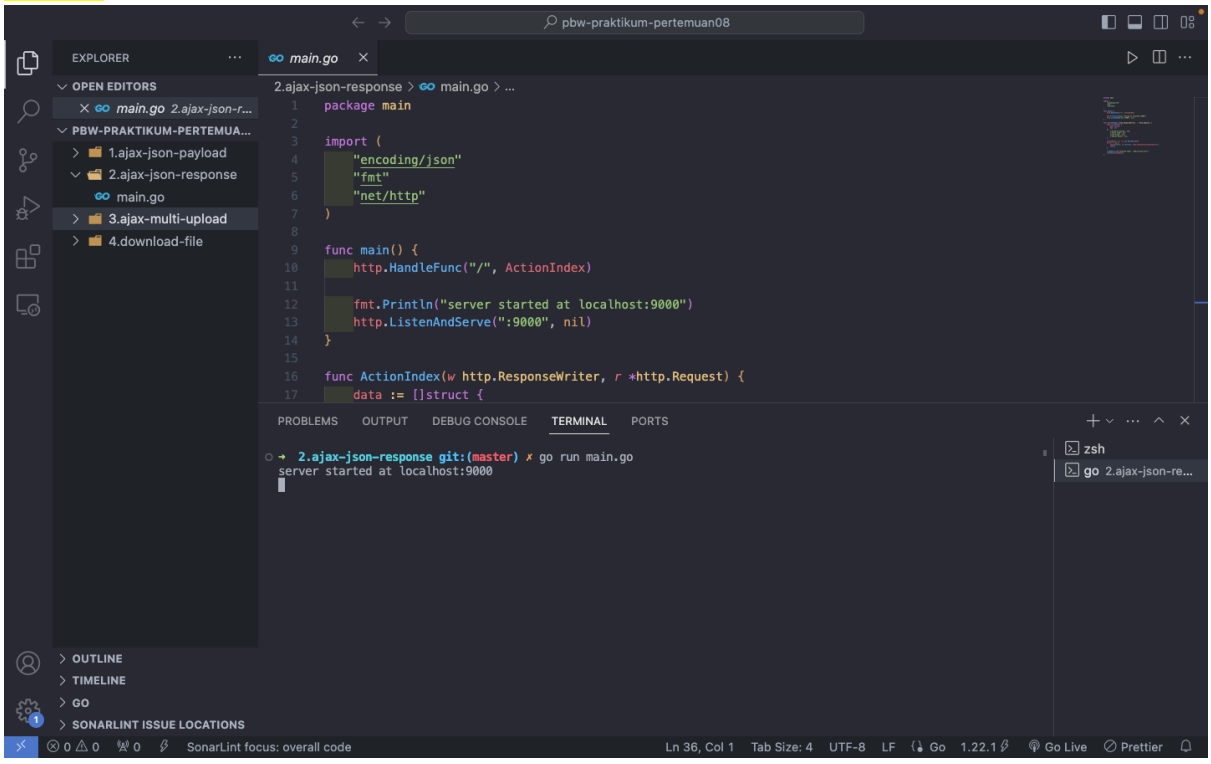


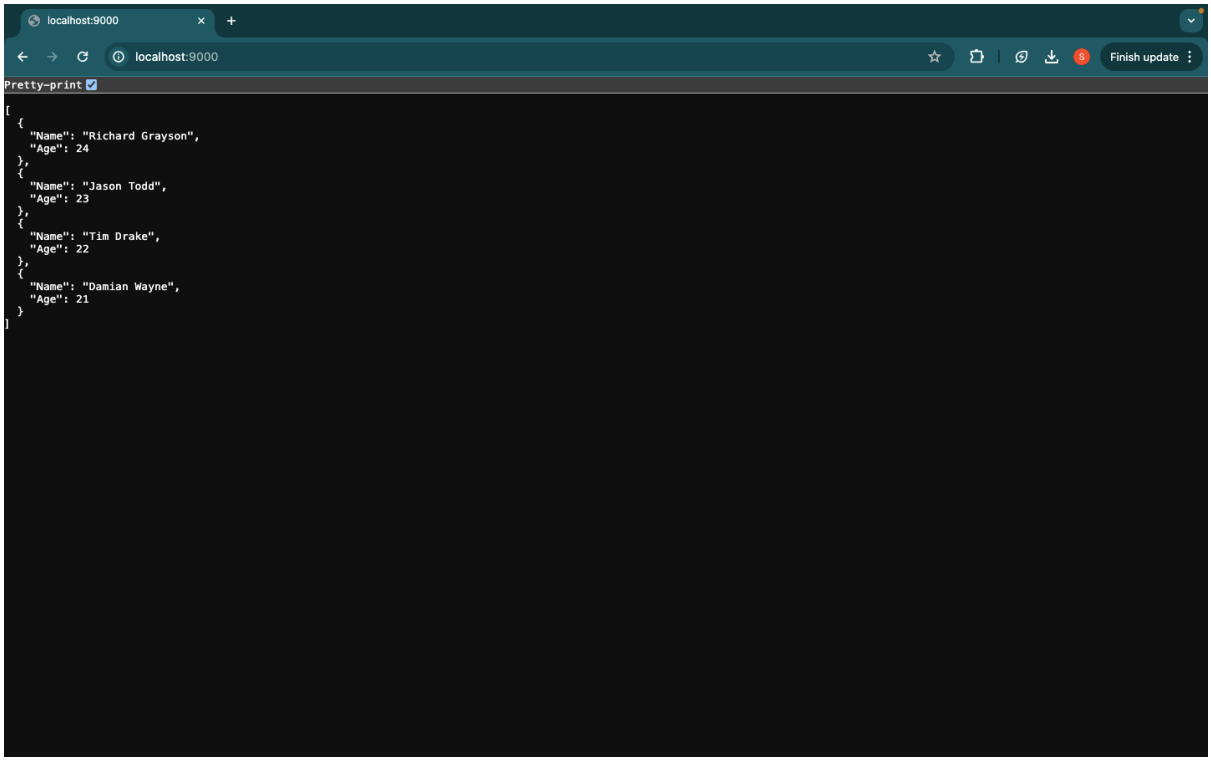
2. AJAX JSON Response

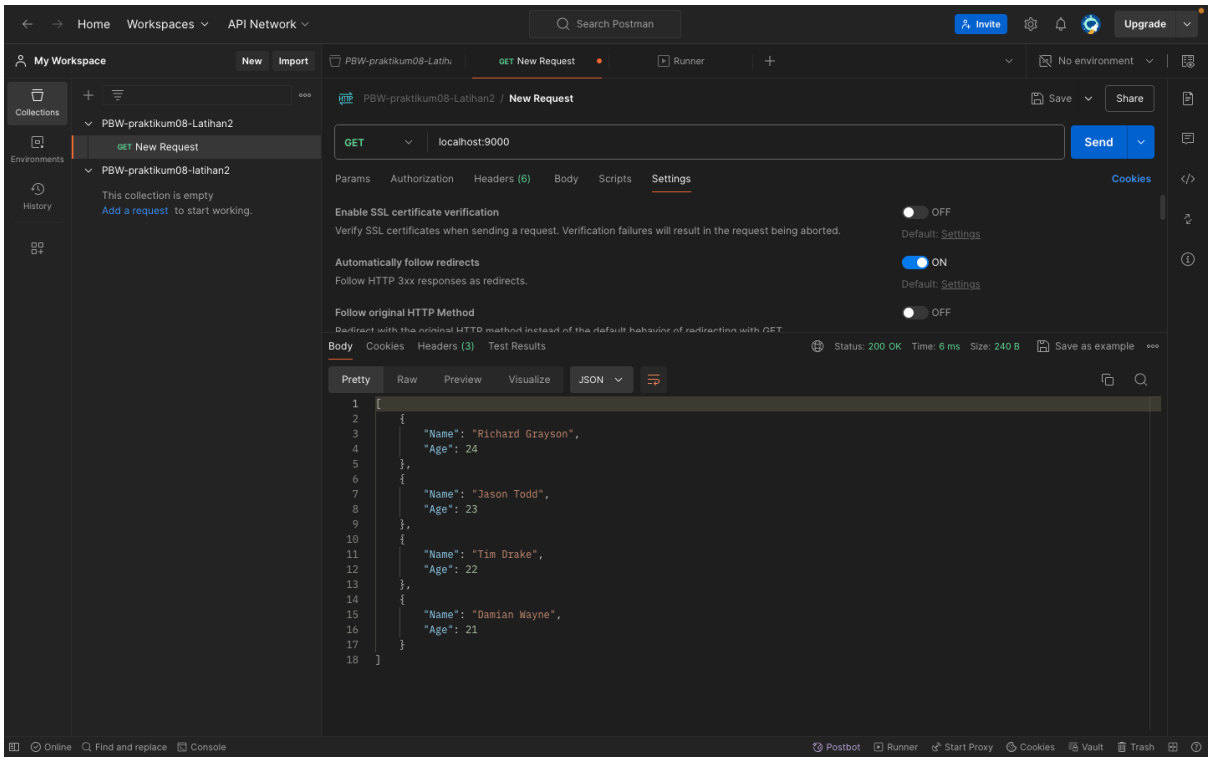
Screenshot Source Code



Output



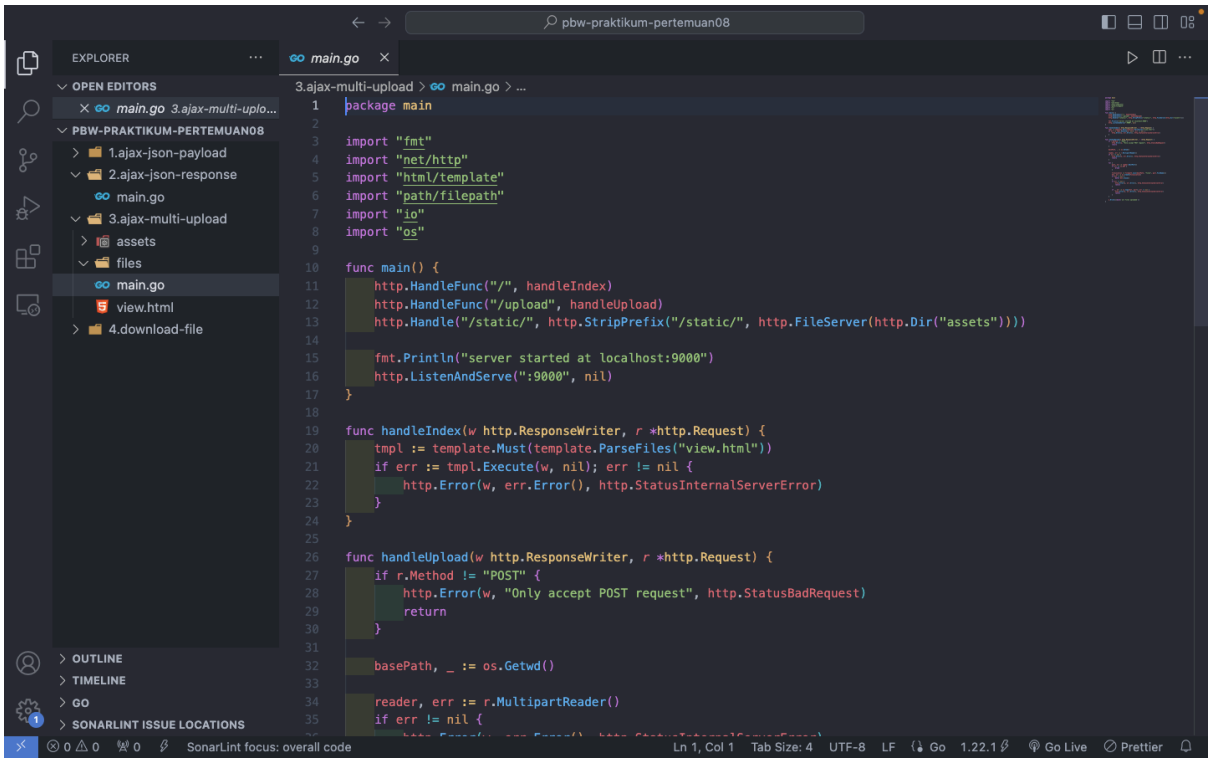

localhost:9000
localhost:9000
Finish update
Pretty-print
[
{
"Name": "Richard Grayson",
"Age": 24
},
{
"Name": "Jason Todd",
"Age": 23
},
{
"Name": "Tim Drake",
"Age": 22
},
{
"Name": "Damian Wayne",
"Age": 21
}
]


Home
Workspaces
API Network
Search Postman
Invite
Upgrade
My Workspace
New
Import
Collections
Environments
History
PBW-praktikum08-Latihan2
GET New Request
PBW-praktikum08-latihan2
This collection is empty
Add a request to start working.
PBW-praktikum08-Latihan2
GET New Request
Runner
No environment
PBW-praktikum08-Latihan2 / New Request
Save
Share
GET
localhost:9000
Send
Params
Authorization
Headers (6)
Body
Scripts
Settings
Cookies
Enable SSL certificate verification
Verify SSL certificates when sending a request. Verification failures will result in the request being aborted.
OFF
Default: Settings
Automatically follow redirects
Follow HTTP 3xx responses as redirects.
ON
Default: Settings
Follow original HTTP Method
OFF
Body
Cookies
Headers (3)
Test Results
Status: 200 OK
Time: 6 ms
Size: 240 B
Save as example
Pretty
Raw
Preview
Visualize
JSON
[
{
"Name": "Richard Grayson",
"Age": 24
},
{
"Name": "Jason Todd",
"Age": 23
},
{
"Name": "Tim Drake",
"Age": 22
},
{
"Name": "Damian Wayne",
"Age": 21
}
]
Online
Find and replace
Console
Postbot
Runner
Start Proxy
Cookies
Vault
Trash

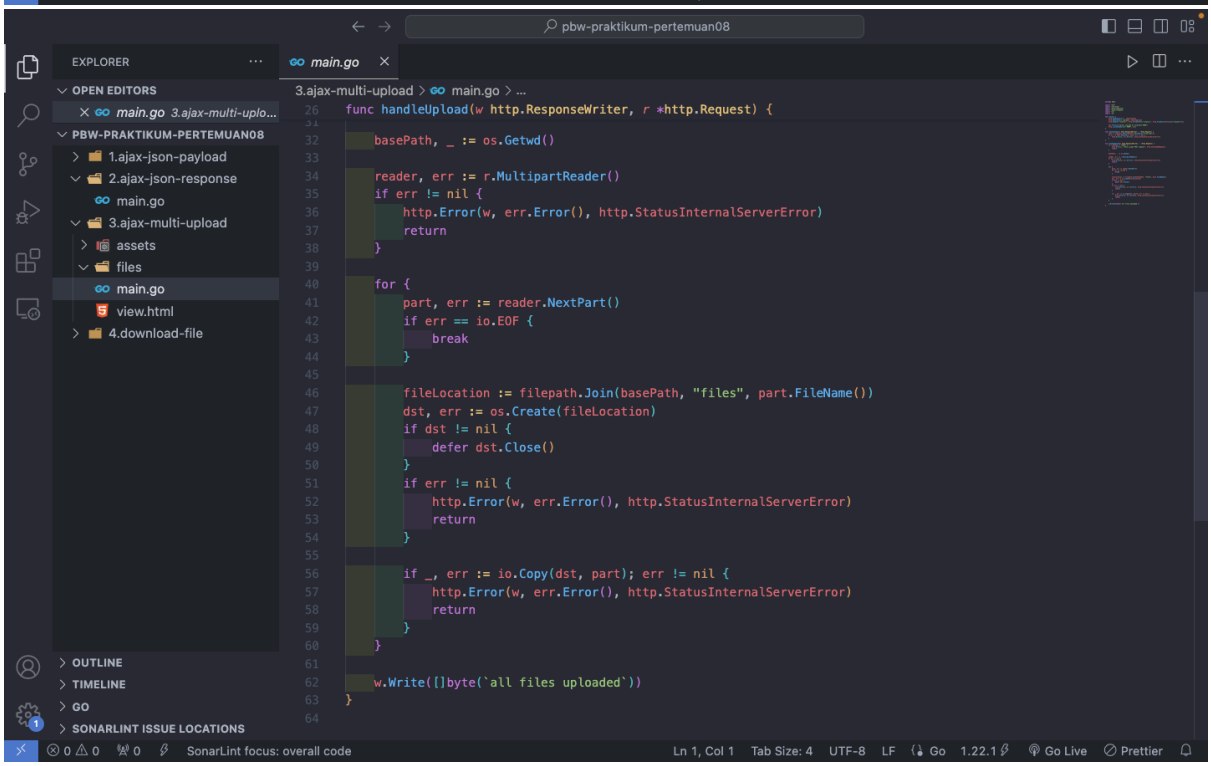
3. AJAX Multiple File Upload

Screenshot Source Code

Main.go

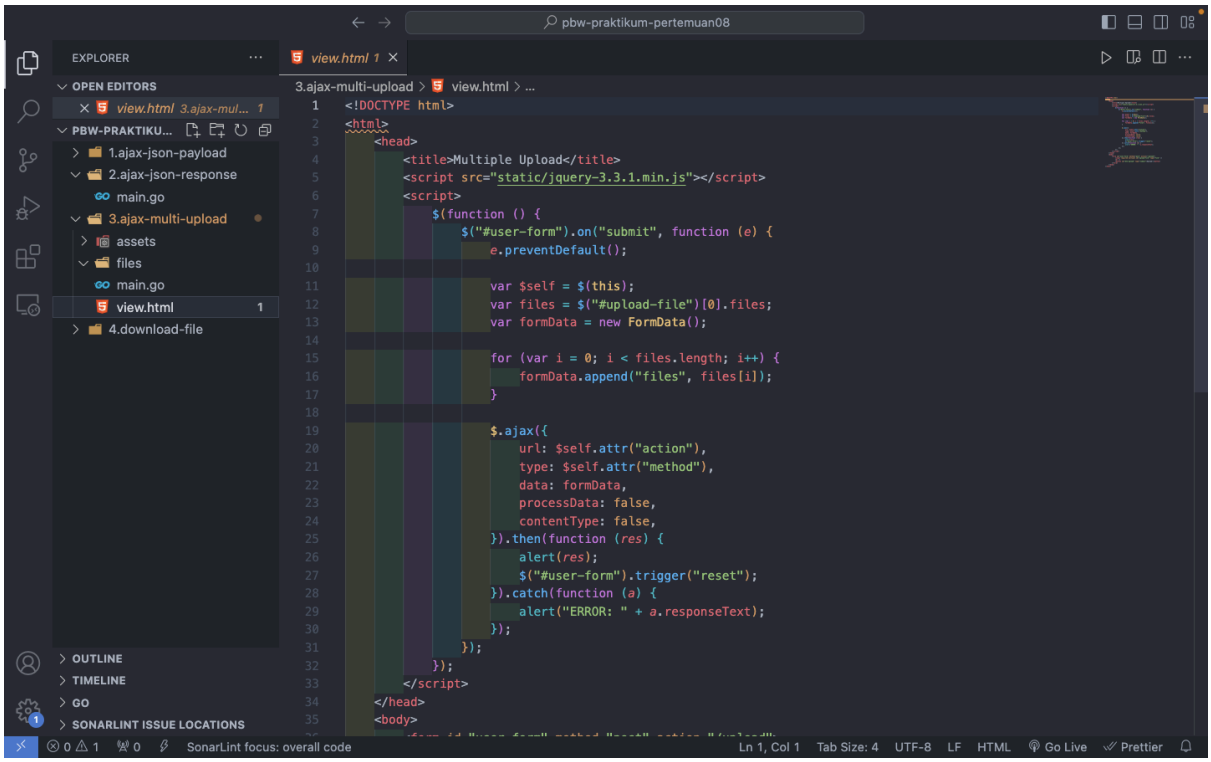


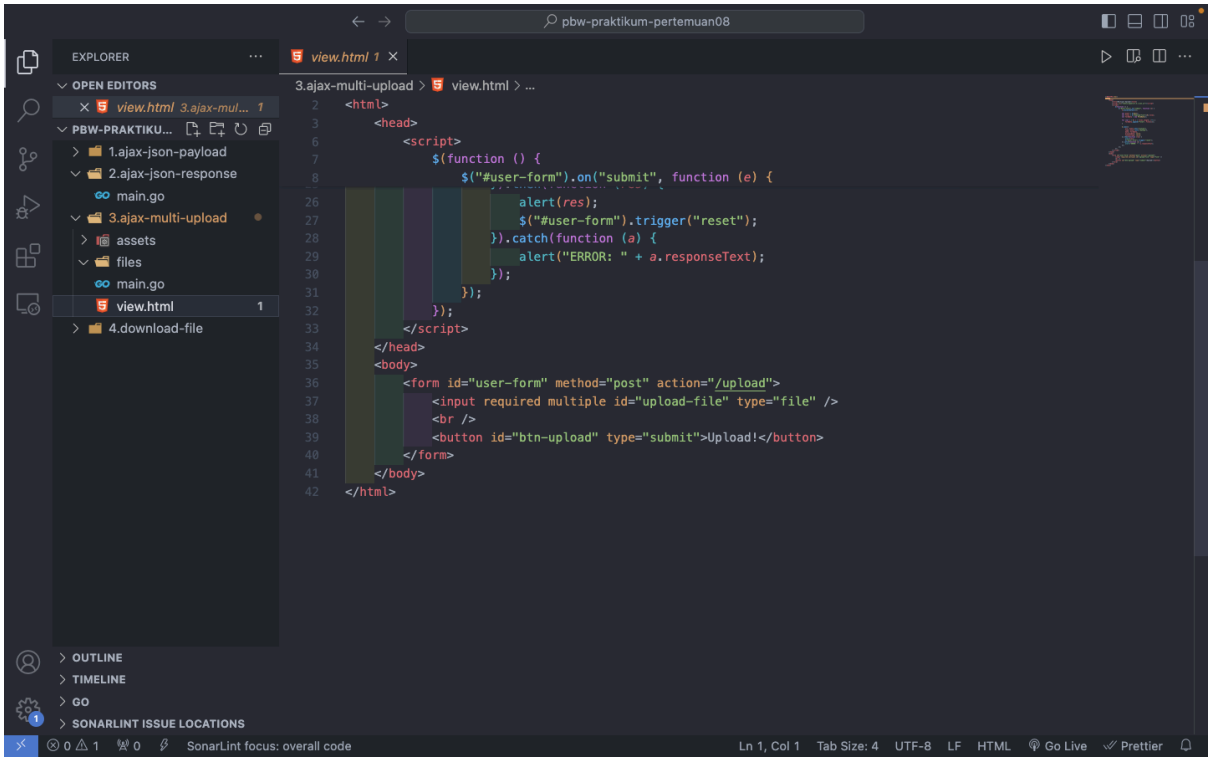
```
package main

import "fmt"
import "net/http"
import "html/template"
import "path/filepath"
import "io"
import "os"

func main() {
    http.HandleFunc("/", handleIndex)
    http.HandleFunc("/upload", handleUpload)
    http.Handle("/static/", http.StripPrefix("/static/", http.FileServer(http.Dir("assets"))))

    fmt.Println("server started at localhost:9000")
    http.ListenAndServe(":9000", nil)
}

func handleIndex(w http.ResponseWriter, r *http.Request) {
    tmpl := template.Must(template.ParseFiles("view.html"))
    if err := tmpl.Execute(w, nil); err != nil {
        http.Error(w, err.Error(), http.StatusInternalServerError)
    }
}

func handleUpload(w http.ResponseWriter, r *http.Request) {
    if r.Method != "POST" {
        http.Error(w, "Only accept POST request", http.StatusBadRequest)
        return
    }

    basePath, _ := os.Getwd()

    reader, err := r.MultipartReader()
    if err != nil {
```



```
func handleUpload(w http.ResponseWriter, r *http.Request) {

    basePath, _ := os.Getwd()

    reader, err := r.MultipartReader()
    if err != nil {
        http.Error(w, err.Error(), http.StatusInternalServerError)
        return
    }

    for {
        part, err := reader.NextPart()
        if err == io.EOF {
            break
        }

        fileLocation := filepath.Join(basePath, "files", part.FileName())
        dst, err := os.Create(fileLocation)
        if dst != nil {
            defer dst.Close()
        }
        if err != nil {
            http.Error(w, err.Error(), http.StatusInternalServerError)
            return
        }

        if _, err := io.Copy(dst, part); err != nil {
            http.Error(w, err.Error(), http.StatusInternalServerError)
            return
        }
    }

    w.Write([]byte(`all files uploaded`))
}
```

View.html

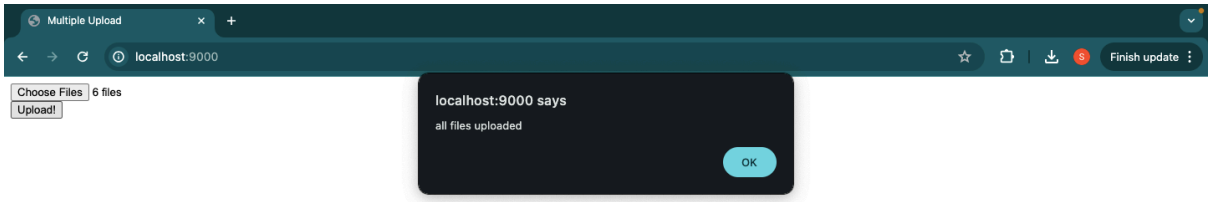

pbw-praktikum-pertemuan08
EXPLORER
view.html 1
OPEN EDITORS
view.html 3.ajax-mul... 1
PBW-PRAKTIKU...
1.ajax-json-payload
2.ajax-json-response
main.go
3.ajax-multi-upload
assets
files
main.go
view.html 1
4.download-file
3.ajax-multi-upload > view.html > ...
<!DOCTYPE html>
<html>
<head>
<title>Multiple Upload</title>
<script src="static/jquery-3.3.1.min.js"></script>
<script>
$(function () {
$("#user-form").on("submit", function (e) {
e.preventDefault();
var $self = $(this);
var files = $("#upload-file")[0].files;
var formData = new FormData();
for (var i = 0; i < files.length; i++) {
formData.append("files", files[i]);
}
$.ajax({
url: $self.attr("action"),
type: $self.attr("method"),
data: formData,
processData: false,
contentType: false,
}).then(function (res) {
alert(res);
$("#user-form").trigger("reset");
}).catch(function (a) {
alert("ERROR: " + a.responseText);
});
});
});
</script>
</head>
<body>
OUTLINE
TIMELINE
GO
SONARLINT ISSUE LOCATIONS
SonarLint focus: overall code
Ln 1, Col 1
Tab Size: 4
UTF-8
LF
HTML
Go Live
Prettier


pbw-praktikum-pertemuan08
EXPLORER
view.html 1
OPEN EDITORS
view.html 3.ajax-mul... 1
PBW-PRAKTIKU...
1.ajax-json-payload
2.ajax-json-response
main.go
3.ajax-multi-upload
assets
files
main.go
view.html 1
4.download-file
3.ajax-multi-upload > view.html > ...
<html>
<head>
<script>
$(function () {
$("#user-form").on("submit", function (e) {
alert(res);
$("#user-form").trigger("reset");
}).catch(function (a) {
alert("ERROR: " + a.responseText);
});
});
});
</script>
</head>
<body>
<form id="user-form" method="post" action="/upload">
<input required multiple id="upload-file" type="file" />

<button id="btn-upload" type="submit">Upload!</button>
</form>
</body>
</html>
OUTLINE
TIMELINE
GO
SONARLINT ISSUE LOCATIONS
SonarLint focus: overall code
Ln 1, Col 1
Tab Size: 4
UTF-8
LF
HTML
Go Live
Prettier

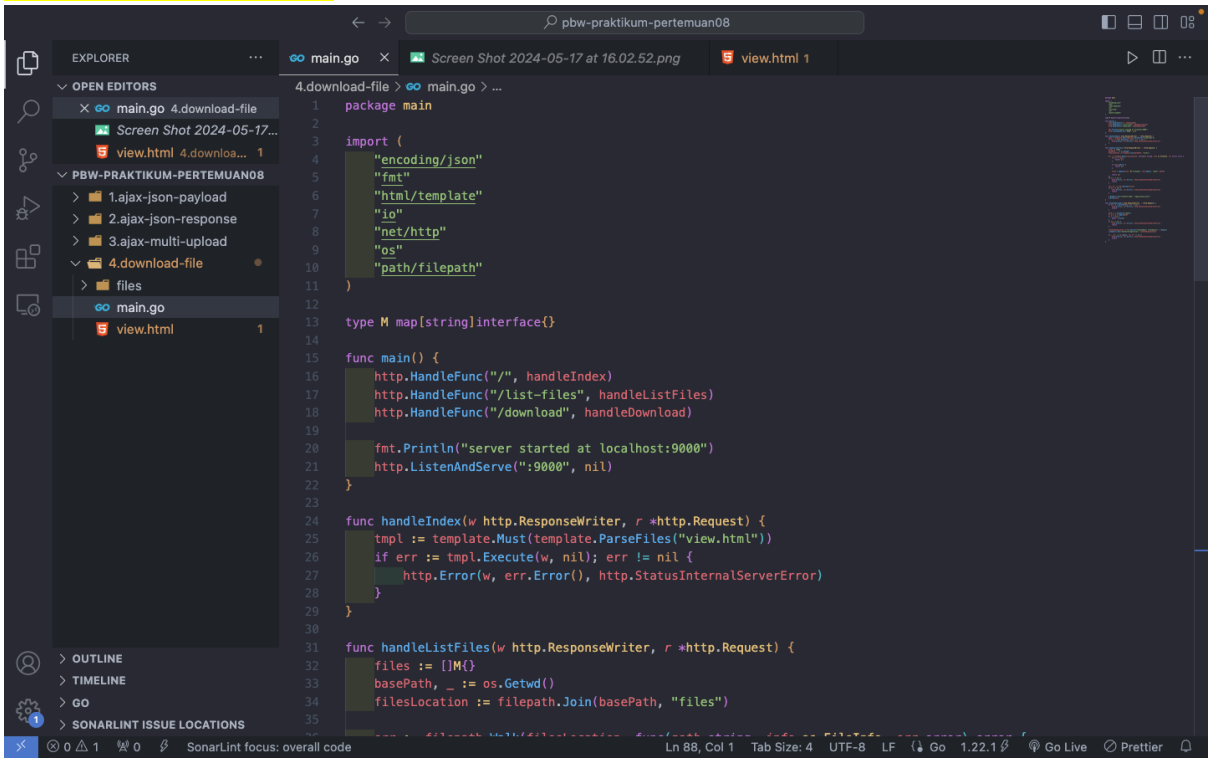
Output


Multiple Upload
localhost:9000
Finish update
Choose Files
No file chosen
Upload!

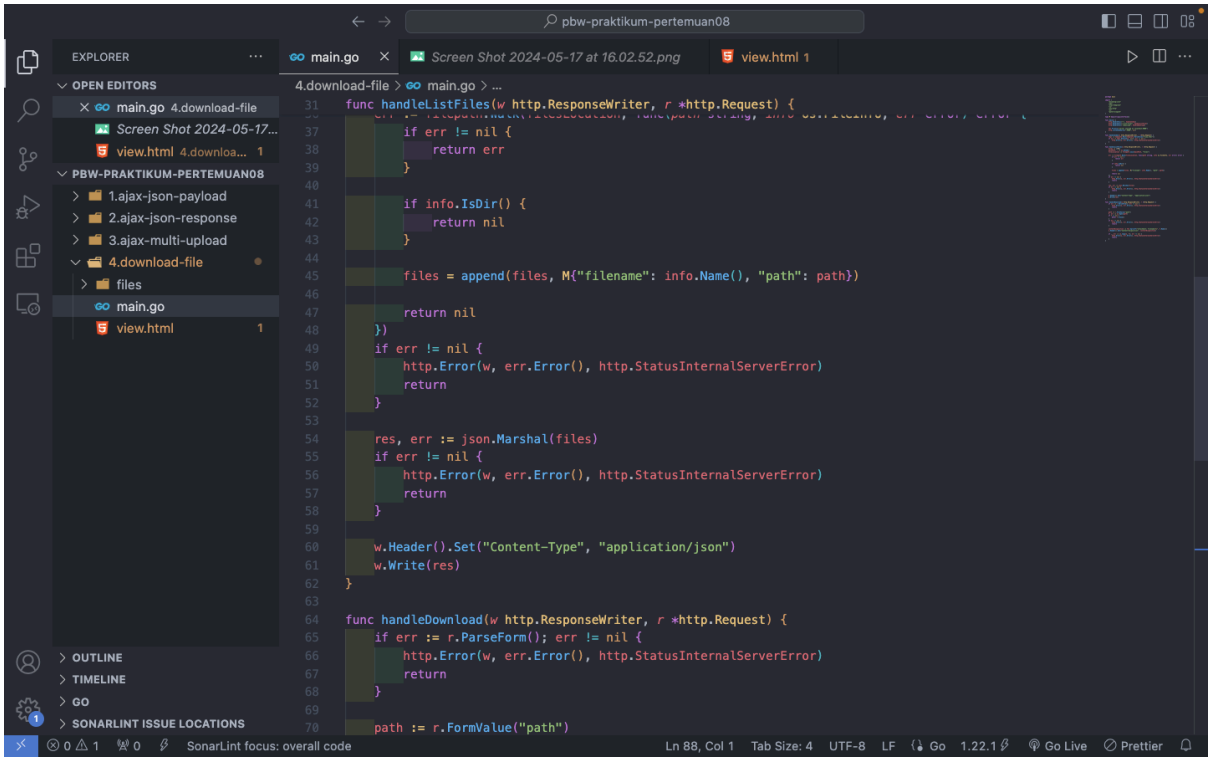

Multiple Upload
localhost:9000
Finish update
Choose Files
6 files
Upload!
localhost:9000 says
all files uploaded
OK

4. Download File

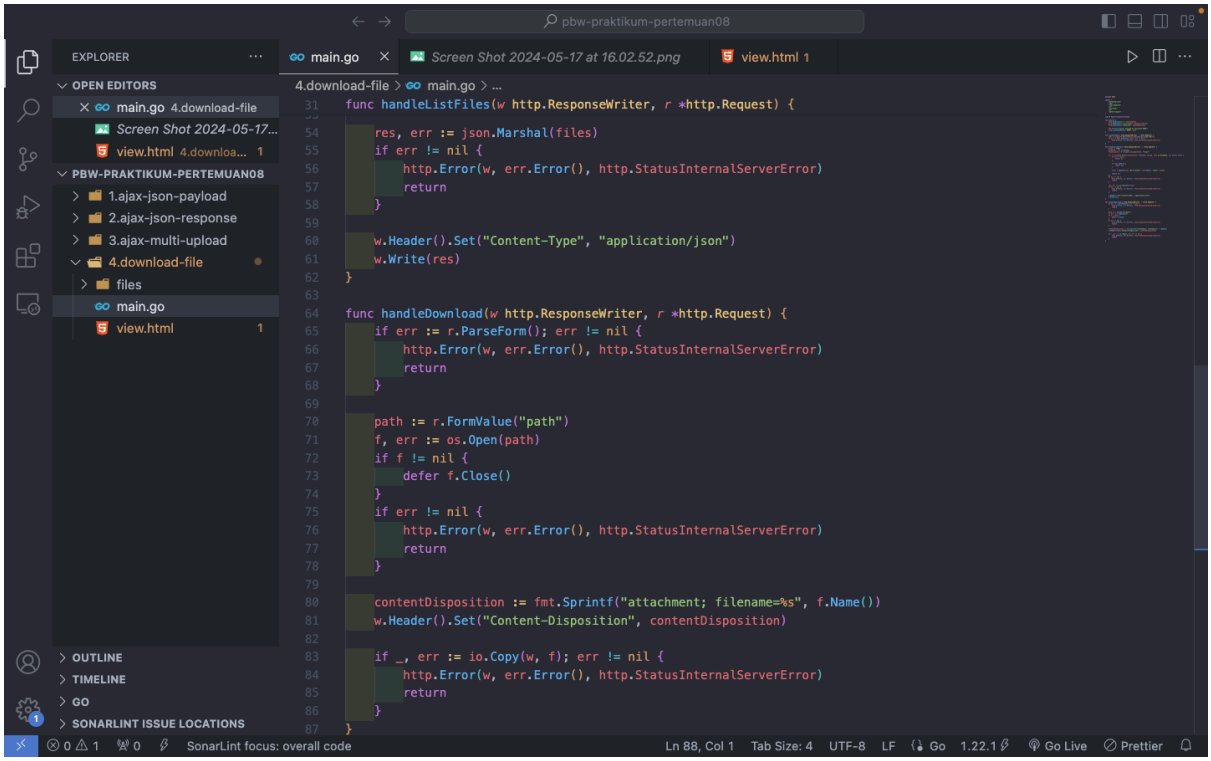
Screenshot Source Code



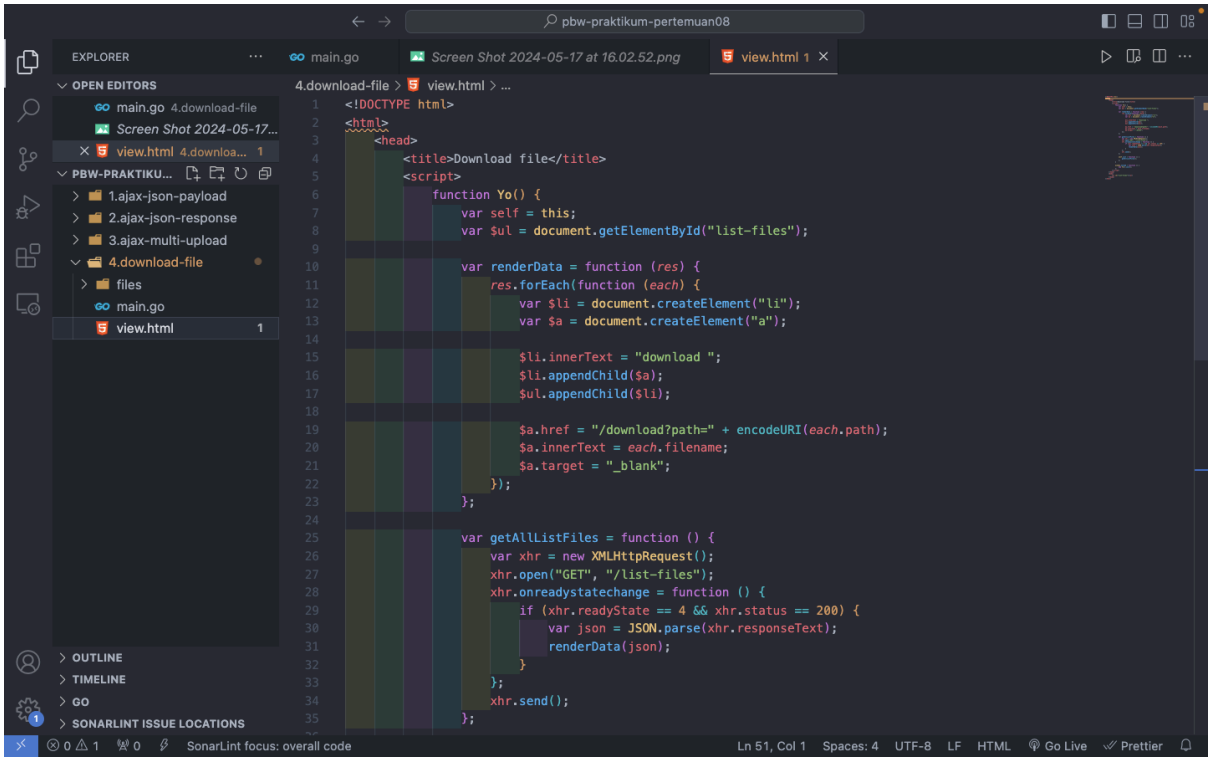
```
package main

import (
    "encoding/json"
    "fmt"
    "html/template"
    "io"
    "net/http"
    "os"
    "path/filepath"
)

type M map[string]interface{}

func main() {
    http.HandleFunc("/", handleIndex)
    http.HandleFunc("/list-files", handleListFiles)
    http.HandleFunc("/download", handleDownload)

    fmt.Println("server started at localhost:9000")
    http.ListenAndServe(":9000", nil)
}

func handleIndex(w http.ResponseWriter, r *http.Request) {
    tmpl := template.Must(template.ParseFiles("view.html"))
    if err := tmpl.Execute(w, nil); err != nil {
        http.Error(w, err.Error(), http.StatusInternalServerError)
    }
}

func handleListFiles(w http.ResponseWriter, r *http.Request) {
    files := []M{}
    basePath, _ := os.Getwd()
    filesLocation := filepath.Join(basePath, "files")
```

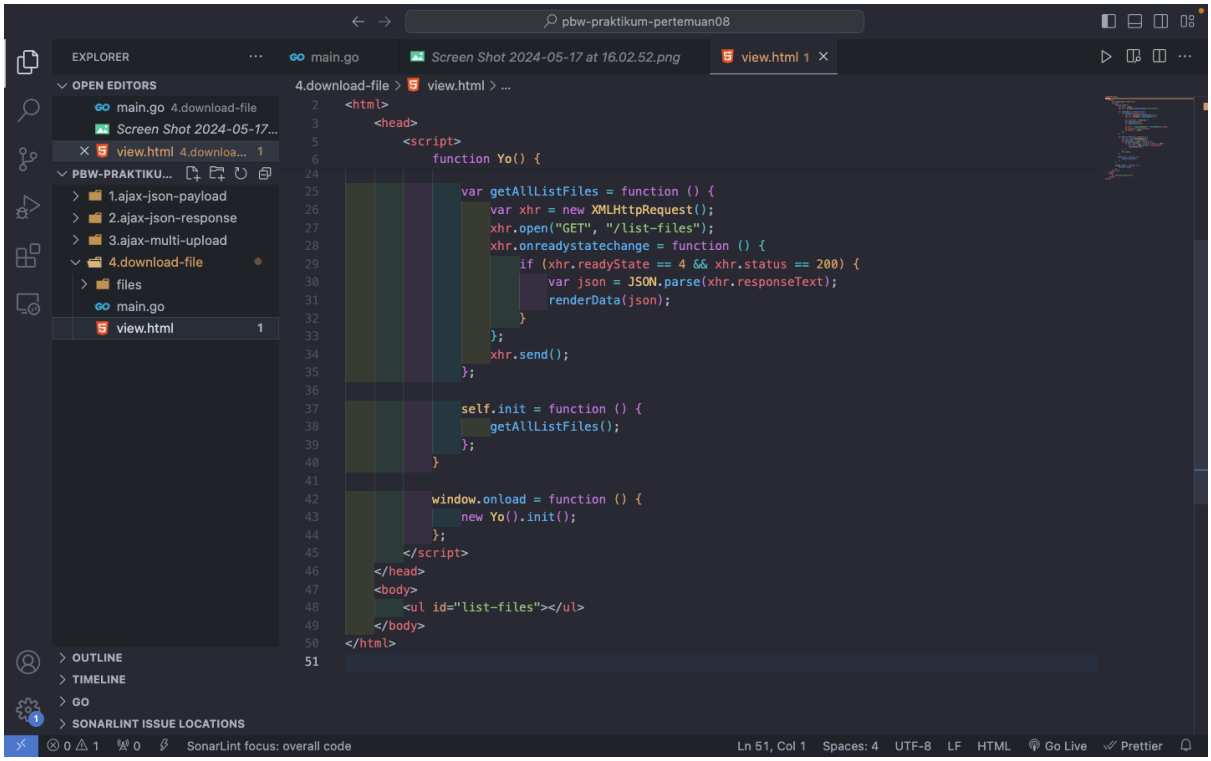


```
func handleListFiles(w http.ResponseWriter, r *http.Request) {
        if err != nil {
            return err
        }

        if info.IsDir() {
            return nil
        }

        files = append(files, M{"filename": info.Name(), "path": path})

        return nil
    })
    if err != nil {
        http.Error(w, err.Error(), http.StatusInternalServerError)
        return
    }

    res, err := json.Marshal(files)
    if err != nil {
        http.Error(w, err.Error(), http.StatusInternalServerError)
        return
    }

    w.Header().Set("Content-Type", "application/json")
    w.Write(res)
}

func handleDownload(w http.ResponseWriter, r *http.Request) {
    if err := r.ParseForm(); err != nil {
        http.Error(w, err.Error(), http.StatusInternalServerError)
        return
    }

    path := r.FormValue("path")
```

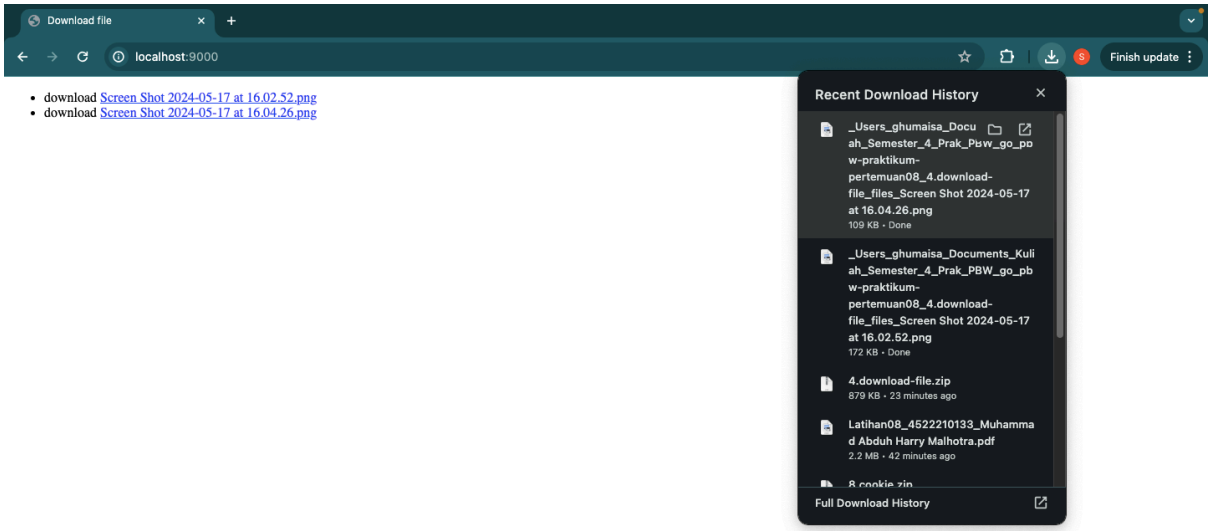

pbw-praktikum-pertemuan08
EXPLORER
main.go
Screen Shot 2024-05-17 at 16.02.52.png
view.html 1
OPEN EDITORS
main.go 4.download-file
Screen Shot 2024-05-17...
view.html 4.downloa... 1
PBW-PRAKTIKUM-PERTEMUAN08
1.ajax-json-payload
2.ajax-json-response
3.ajax-multi-upload
4.download-file
files
main.go
view.html 1
4.download-file > main.go > ...
func handleListFiles(w http.ResponseWriter, r *http.Request) {
res, err := json.Marshal(files)
if err != nil {
http.Error(w, err.Error(), http.StatusInternalServerError)
return
}
w.Header().Set("Content-Type", "application/json")
w.Write(res)
}
func handleDownload(w http.ResponseWriter, r *http.Request) {
if err := r.ParseForm(); err != nil {
http.Error(w, err.Error(), http.StatusInternalServerError)
return
}
path := r.FormValue("path")
f, err := os.Open(path)
if f != nil {
defer f.Close()
}
if err != nil {
http.Error(w, err.Error(), http.StatusInternalServerError)
return
}
contentDisposition := fmt.Sprintf("attachment; filename=%s", f.Name())
w.Header().Set("Content-Disposition", contentDisposition)
if _, err := io.Copy(w, f); err != nil {
http.Error(w, err.Error(), http.StatusInternalServerError)
return
}
}
OUTLINE
TIMELINE
GO
SONARLINT ISSUE LOCATIONS
SonarLint focus: overall code
Ln 88, Col 1
Tab Size: 4
UTF-8
LF
Go
1.22.1
Go Live
Prettier

View.html


pbw-praktikum-pertemuan08
EXPLORER
main.go
Screen Shot 2024-05-17 at 16.02.52.png
view.html 1
OPEN EDITORS
main.go 4.download-file
Screen Shot 2024-05-17...
view.html 4.downloa... 1
PBW-PRAKTIKU...
1.ajax-json-payload
2.ajax-json-response
3.ajax-multi-upload
4.download-file
files
main.go
view.html 1
4.download-file > view.html > ...
<!DOCTYPE html>
<html>
<head>
<title>Download file</title>
<script>
function Yo() {
var self = this;
var $ul = document.getElementById("list-files");
var renderData = function (res) {
res.forEach(function (each) {
var $li = document.createElement("li");
var $a = document.createElement("a");
$li.innerText = "download ";
$li.appendChild($a);
$ul.appendChild($li);
$a.href = "/download?path=" + encodeURI(each.path);
$a.innerText = each.filename;
$a.target = "_blank";
});
};
var getAllListFiles = function () {
var xhr = new XMLHttpRequest();
xhr.open("GET", "/list-files");
xhr.onreadystatechange = function () {
if (xhr.readyState == 4 && xhr.status == 200) {
var json = JSON.parse(xhr.responseText);
renderData(json);
}
};
xhr.send();
};
OUTLINE
TIMELINE
GO
SONARLINT ISSUE LOCATIONS
SonarLint focus: overall code
Ln 51, Col 1
Spaces: 4
UTF-8
LF
HTML
Go Live
Prettier

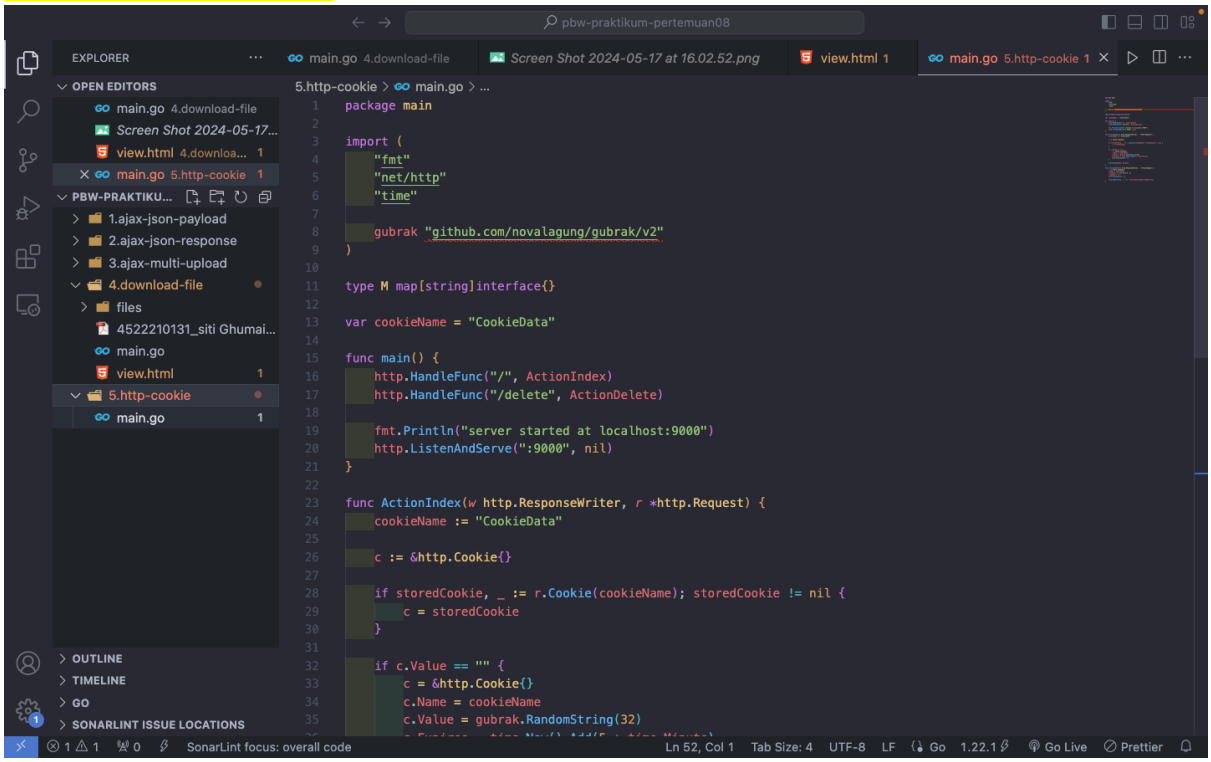

pbw-praktikum-pertemuan08
EXPLORER
main.go
Screen Shot 2024-05-17 at 16.02.52.png
view.html 1
OPEN EDITORS
main.go 4.download-file
Screen Shot 2024-05-17...
view.html 4.downloa... 1
PBW-PRAKTIKU...
1.ajax-json-payload
2.ajax-json-response
3.ajax-multi-upload
4.download-file
files
main.go
view.html 1
4.download-file > view.html > ...
<html>
<head>
<script>
function Yo() {
var getAllListFiles = function () {
var xhr = new XMLHttpRequest();
xhr.open("GET", "/list-files");
xhr.onreadystatechange = function () {
if (xhr.readyState == 4 && xhr.status == 200) {
var json = JSON.parse(xhr.responseText);
renderData(json);
}
};
xhr.send();
};
self.init = function () {
getAllListFiles();
};
}
window.onload = function () {
new Yo().init();
};
</script>
</head>
<body>
<ul id="list-files">
</body>
</html>
OUTLINE
TIMELINE
GO
SONARLINT ISSUE LOCATIONS
SonarLint focus: overall code
Ln 51, Col 1
Spaces: 4
UTF-8
LF
HTML
Go Live
Prettier

Output

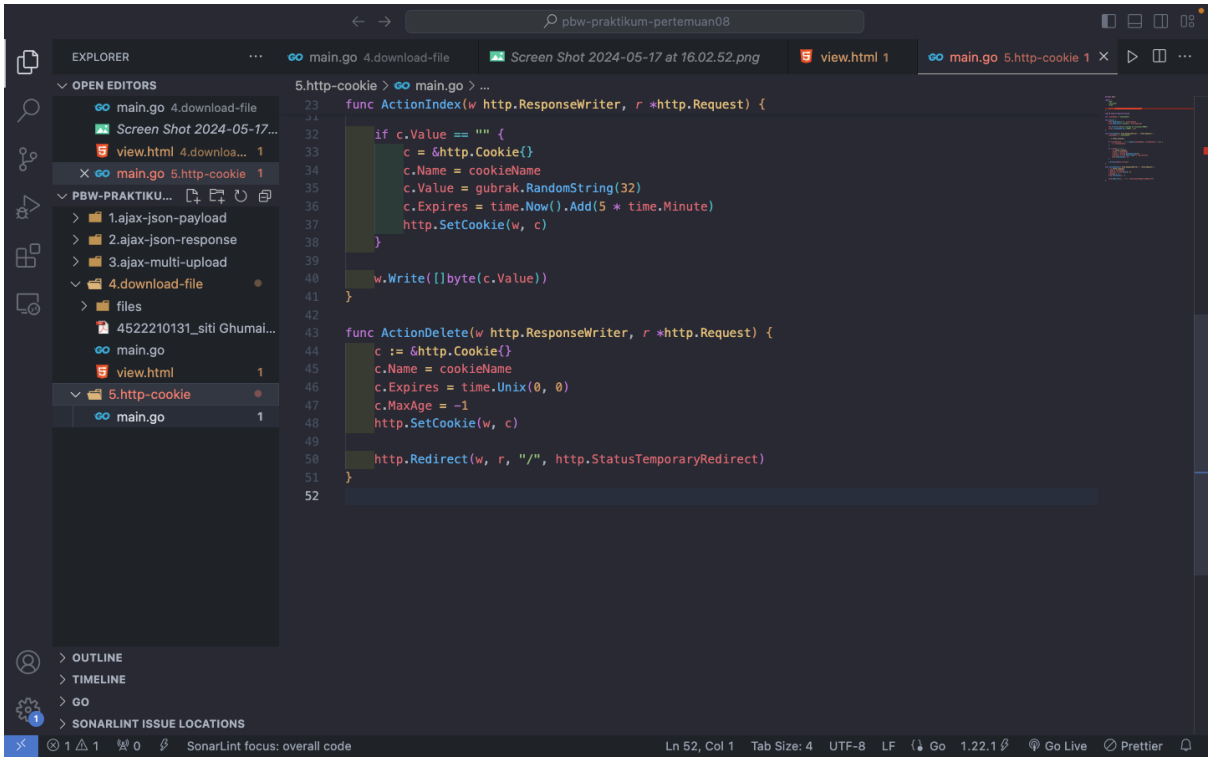

Download file
localhost:9000
Finish update
download Screen Shot 2024-05-17 at 16.02.52.png
download Screen Shot 2024-05-17 at 16.04.26.png
Recent Download History
_Users_ghumaisa_Documah_Semester_4_Prak_Pbw_go_pbw-praktikum-pertemuan08_4.download-file_files_Screen Shot 2024-05-17 at 16.04.26.png
109 KB • Done
_Users_ghumaisa_Documents_Kuliah_Semester_4_Prak_PBW_go_pbw-praktikum-pertemuan08_4.download-file_files_Screen Shot 2024-05-17 at 16.02.52.png
172 KB • Done
4.download-file.zip
879 KB • 23 minutes ago
Latihan08_4522210133_Muhammad Abduh Harry Malhotra.pdf
2.2 MB • 42 minutes ago
Full Download History

5. HTTP Cookie

Screenshot Source Code

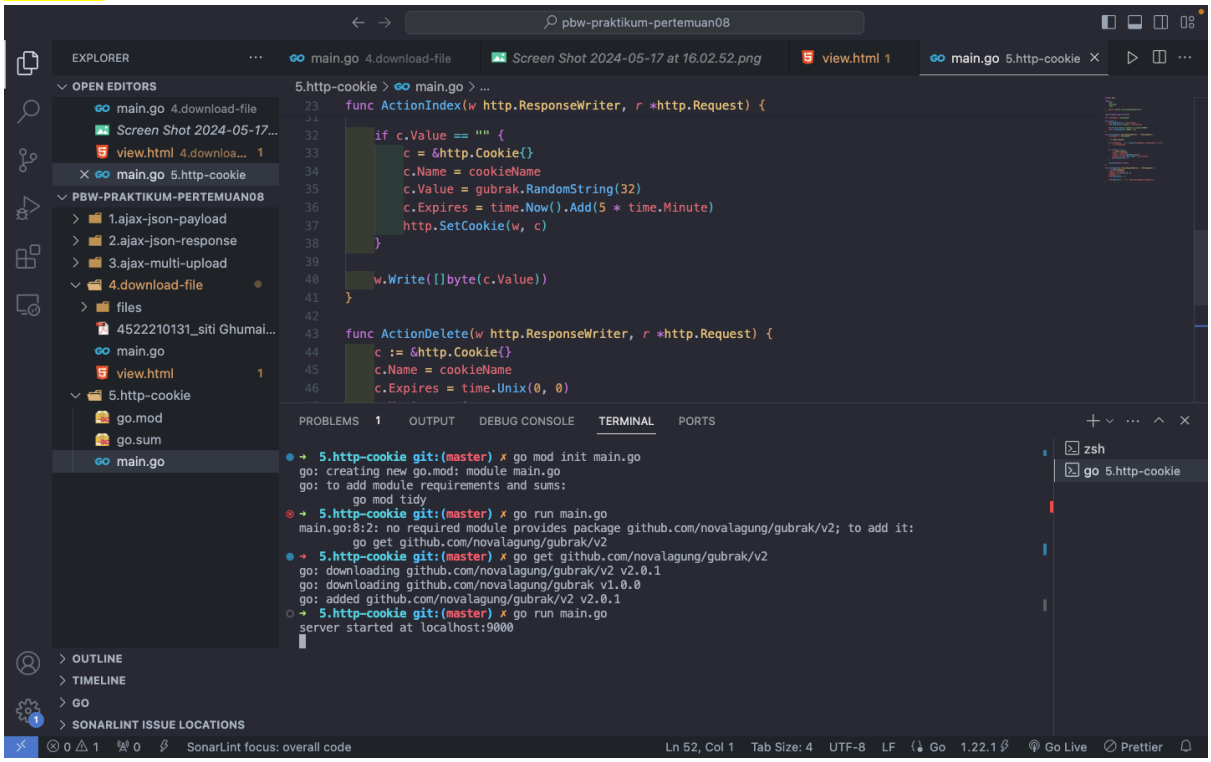


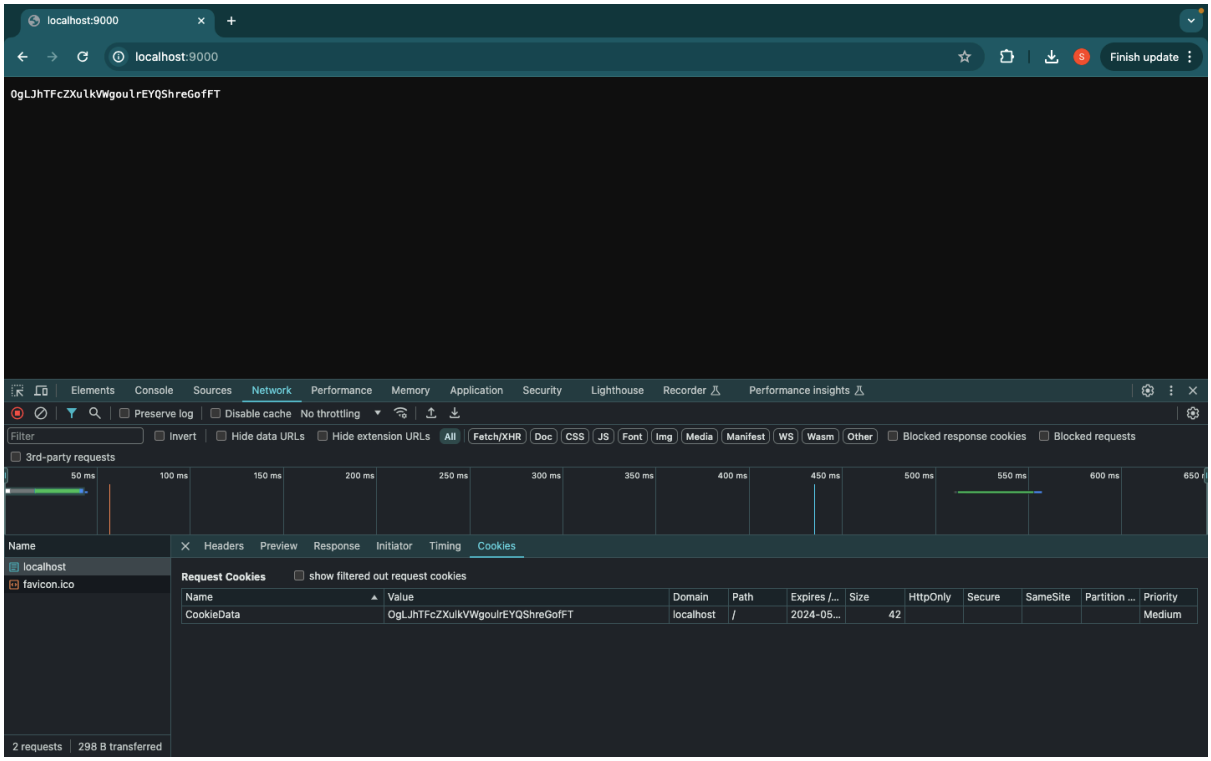
```
package main

import (
    "fmt"
    "net/http"
    "time"

    gubrak "github.com/novalagung/gubrak/v2"
)

type M map[string]interface{}

var cookieName = "CookieData"

func main() {
    http.HandleFunc("/", ActionIndex)
    http.HandleFunc("/delete", ActionDelete)

    fmt.Println("server started at localhost:9000")
    http.ListenAndServe(":9000", nil)
}

func ActionIndex(w http.ResponseWriter, r *http.Request) {
    cookieName := "CookieData"

    c := &http.Cookie{}

    if storedCookie, _ := r.Cookie(cookieName); storedCookie != nil {
        c = storedCookie
    }

    if c.Value == "" {
        c = &http.Cookie{}
        c.Name = cookieName
        c.Value = gubrak.RandomString(32)
```

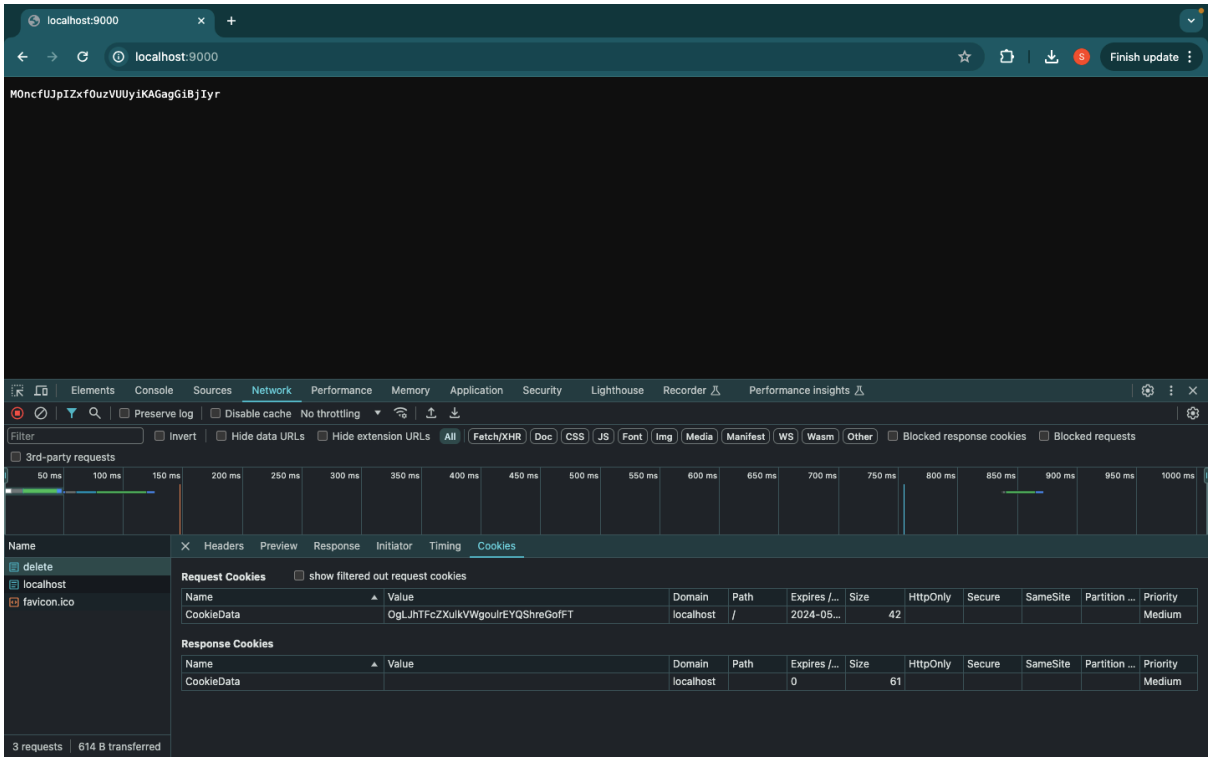


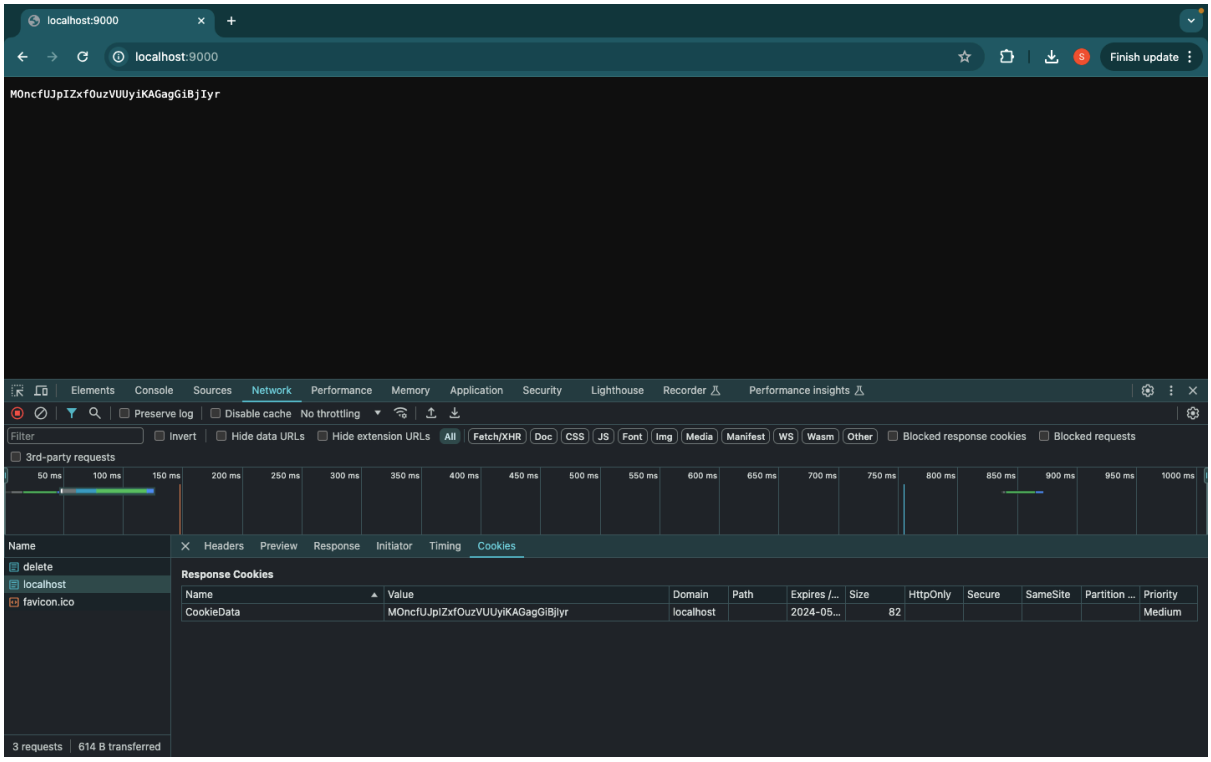
```
func ActionIndex(w http.ResponseWriter, r *http.Request) {

    if c.Value == "" {
        c = &http.Cookie{}
        c.Name = cookieName
        c.Value = gubrak.RandomString(32)
        c.Expires = time.Now().Add(5 * time.Minute)
        http.SetCookie(w, c)
    }

    w.Write([]byte(c.Value))
}

func ActionDelete(w http.ResponseWriter, r *http.Request) {
    c := &http.Cookie{}
    c.Name = cookieName
    c.Expires = time.Unix(0, 0)
    c.MaxAge = -1
    http.SetCookie(w, c)

    http.Redirect(w, r, "/", http.StatusTemporaryRedirect)
}
```

Output


pbw-praktikum-pertemuan08
EXPLORER
main.go 4.download-file
Screen Shot 2024-05-17 at 16.02.52.png
view.html 1
main.go 5.http-cookie
OPEN EDITORS
PBW-PRAKTIKUM-PERTEMUAN08
1.ajax-json-payload
2.ajax-json-response
3.ajax-multi-upload
4.download-file
files
5.http-cookie
go.mod
go.sum
main.go
view.html
func ActionIndex(w http.ResponseWriter, r *http.Request) {
if c.Value == "" {
c = &http.Cookie{}
c.Name = cookieName
c.Value = gubrak.RandomString(32)
c.Expires = time.Now().Add(5 * time.Minute)
http.SetCookie(w, c)
}
w.Write([]byte(c.Value))
}
func ActionDelete(w http.ResponseWriter, r *http.Request) {
c := &http.Cookie{}
c.Name = cookieName
c.Expires = time.Unix(0, 0)
PROBLEMS 1
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
5.http-cookie git:(master) ✗ go mod init main.go
go: creating new go.mod: module main.go
go: to add module requirements and sums:
go mod tidy
5.http-cookie git:(master) ✗ go run main.go
main.go:8:2: no required module provides package github.com/novalagung/gubrak/v2; to add it:
go get github.com/novalagung/gubrak/v2
5.http-cookie git:(master) ✗ go get github.com/novalagung/gubrak/v2
go: downloading github.com/novalagung/gubrak/v2 v2.0.1
go: downloading github.com/novalagung/gubrak v1.0.0
go: added github.com/novalagung/gubrak/v2 v2.0.1
5.http-cookie git:(master) ✗ go run main.go
server started at localhost:9000
zsh
go 5.http-cookie
OUTLINE
TIMELINE
GO
SONARLINT ISSUE LOCATIONS
SonarLint focus: overall code
Ln 52, Col 1
Tab Size: 4
UTF-8
LF
Go
1.22.1
Go Live
Prettier


localhost:9000
Finish update
OgLJhTFcZXulkVWgoulrEYQShreGofFT
Elements
Console
Sources
Network
Performance
Memory
Application
Security
Lighthouse
Recorder
Performance insights
Preserve log
Disable cache
No throttling
Filter
Invert
Hide data URLs
Hide extension URLs
All
Fetch/XHR
Doc
CSS
JS
Font
Img
Media
Manifest
WS
Wasm
Other
Blocked response cookies
Blocked requests
3rd-party requests
50 ms
100 ms
150 ms
200 ms
250 ms
300 ms
350 ms
400 ms
450 ms
500 ms
550 ms
600 ms
650 ms
Name
localhost
favicon.ico
Headers
Preview
Response
Initiator
Timing
Cookies
Request Cookies
show filtered out request cookies
Name
Value
Domain
Path
Expires /...
Size
HttpOnly
Secure
SameSite
Partition ...
Priority
CookieData
OgLJhTFcZXulkVWgoulrEYQShreGofFT
localhost
/
2024-05...
42
Medium
2 requests
298 B transferred





6. Pertanyaan

a. Apa itu AJAX?

- AJAX (Asynchronous JavaScript and XML) adalah teknik pengembangan web yang memungkinkan halaman web untuk berkomunikasi dengan server di belakang layar tanpa harus me-reload seluruh halaman. Dengan menggunakan AJAX, data dapat diambil atau dikirim ke server secara asinkron, sehingga pengguna dapat terus

berinteraksi dengan halaman web tanpa adanya gangguan atau penundaan yang disebabkan oleh muatan ulang halaman secara keseluruhan. Contohnya, ketika kita mengisi formulir online dan ada bagian dari formulir yang langsung memvalidasi data yang Anda masukkan (seperti email) tanpa perlu memuat ulang halaman, itu biasanya menggunakan AJAX.

b. Apa itu JSON? Apa kegunaannya?
   - **JSON** (JavaScript Object Notation) : format pertukaran data yang ringan dan mudah dibaca serta ditulis oleh manusia, dan mudah diurai serta dihasilkan oleh mesin. JSON digunakan untuk menyimpan dan mentransfer data, terutama antara server dan aplikasi web, sebagai alternatif yang lebih sederhana dibandingkan dengan XML. Kegunaanya,
     1. **Pertukaran Data:** JSON sering digunakan dalam komunikasi antara klien dan server. Misalnya, saat aplikasi web mengambil data dari server menggunakan AJAX, data sering dikirim dalam format JSON.
     2. **Penyimpanan Data:** JSON juga dapat digunakan untuk menyimpan data dalam file atau database. Banyak database NoSQL, seperti MongoDB, menggunakan JSON atau format serupa untuk menyimpan data.
     3. **Konfigurasi:** Beberapa aplikasi menggunakan file konfigurasi dalam format JSON karena mudah dibaca dan dimodifikasi.

c. Apa itu ParseMultipartForm dan MultipartReader ? Jelaskan perbedaannya!
   - **ParseMultipartForm** : fungsi atau metode yang digunakan untuk mem-parsing (menganalisis) data form multipart yang dikirimkan oleh klien (biasanya melalui form HTML dengan metode POST yang berisi file dan data teks). Fungsi ini akan mengekstrak semua bagian dari form (baik file maupun field teks) dan menyimpannya dalam struktur data yang mudah diakses oleh aplikasi. Misalnya, di bahasa pemrograman seperti Go, ParseMultipartForm adalah metode dari objek Request yang mengurai data form multipart dan menempatkannya dalam memori atau di file sementara.
   - **MultipartReader** : objek atau kelas yang menyediakan cara membaca data form multipart secara bertahap (streaming) daripada menguraikannya sekaligus. Ini sangat berguna ketika bekerja dengan file yang sangat besar atau ketika ingin memproses bagian form satu per satu untuk menghemat memori. MultipartReader memungkinkan aplikasi untuk membaca bagian dari form satu per satu, yang bisa sangat efisien dalam hal penggunaan memori dan performa.

d. Apa itu JavaScript? Jelaskan kegunaannya dalam frontend !
   - JavaScript adalah bahasa pemrograman yang digunakan untuk membuat halaman web interaktif dan dinamis. Dalam frontend, JavaScript memungkinkan manipulasi konten HTML dan CSS secara langsung, membuat animasi, validasi form, pembuatan menu dropdown, dan galeri gambar interaktif. Selain itu, JavaScript dapat mengambil data dari server tanpa perlu memuat ulang halaman menggunakan AJAX, serta mendukung pengembangan aplikasi web kompleks dengan framework

seperti React, Angular, dan Vue.js. JavaScript membuat halaman web lebih responsif dan user-friendly.

e. Jelaskan apa itu EventHandler!
   - EventHandler adalah fungsi atau kelas dalam pemrograman web yang bertanggung jawab untuk menangani peristiwa-peristiwa yang terjadi di halaman web, seperti klik tombol, pengisian formulir, pergerakan mouse, atau menekan tombol keyboard. Dengan menambahkan EventHandler ke elemen HTML tertentu, pengembang dapat menentukan tindakan apa yang harus dilakukan ketika suatu peristiwa terjadi, memungkinkan pengalaman pengguna menjadi lebih interaktif dan responsif.

Screenshot push Git-hub