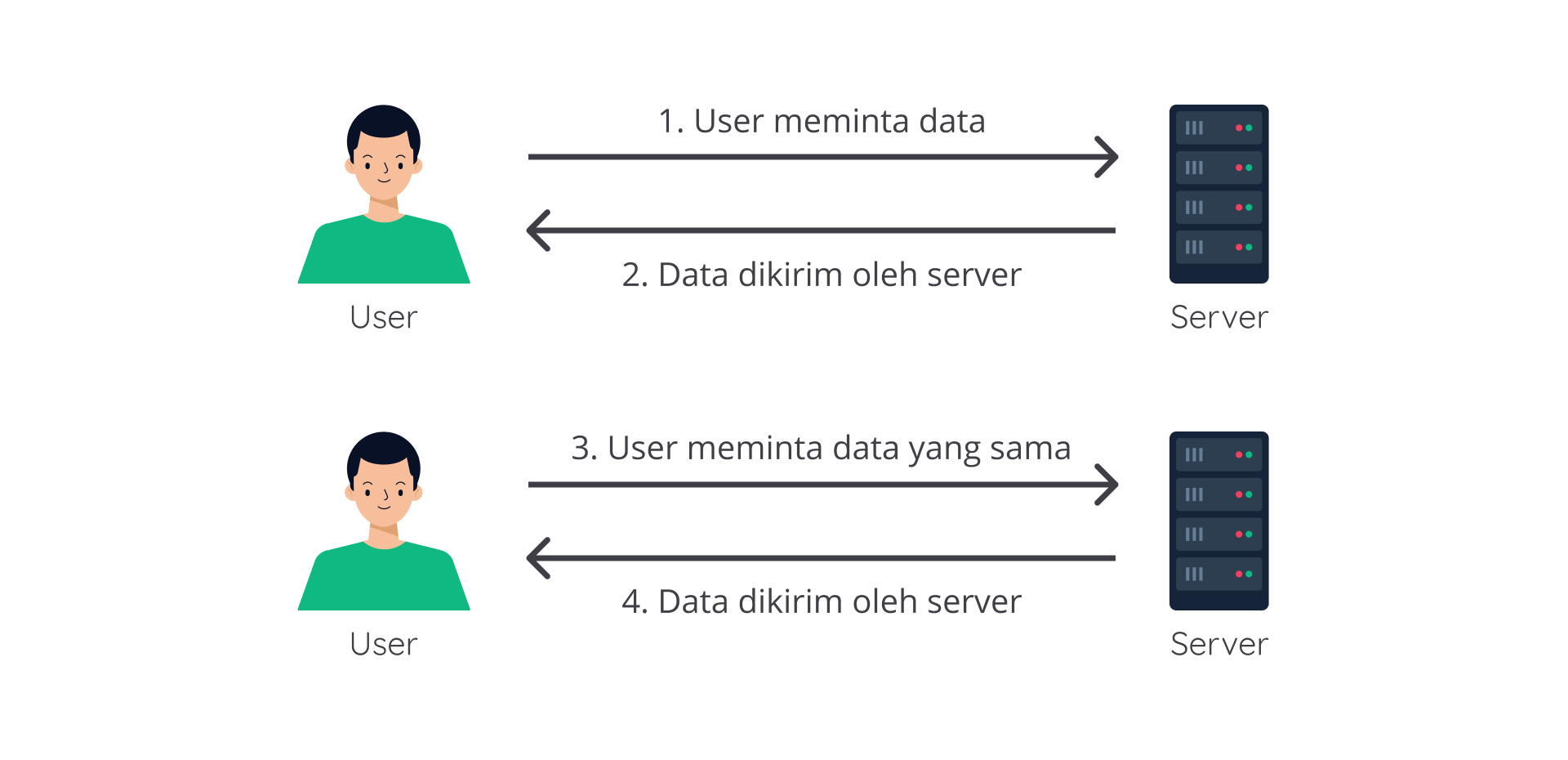
**WEB STORAGE**

***Pengertian dan Fungsi Web Storage***

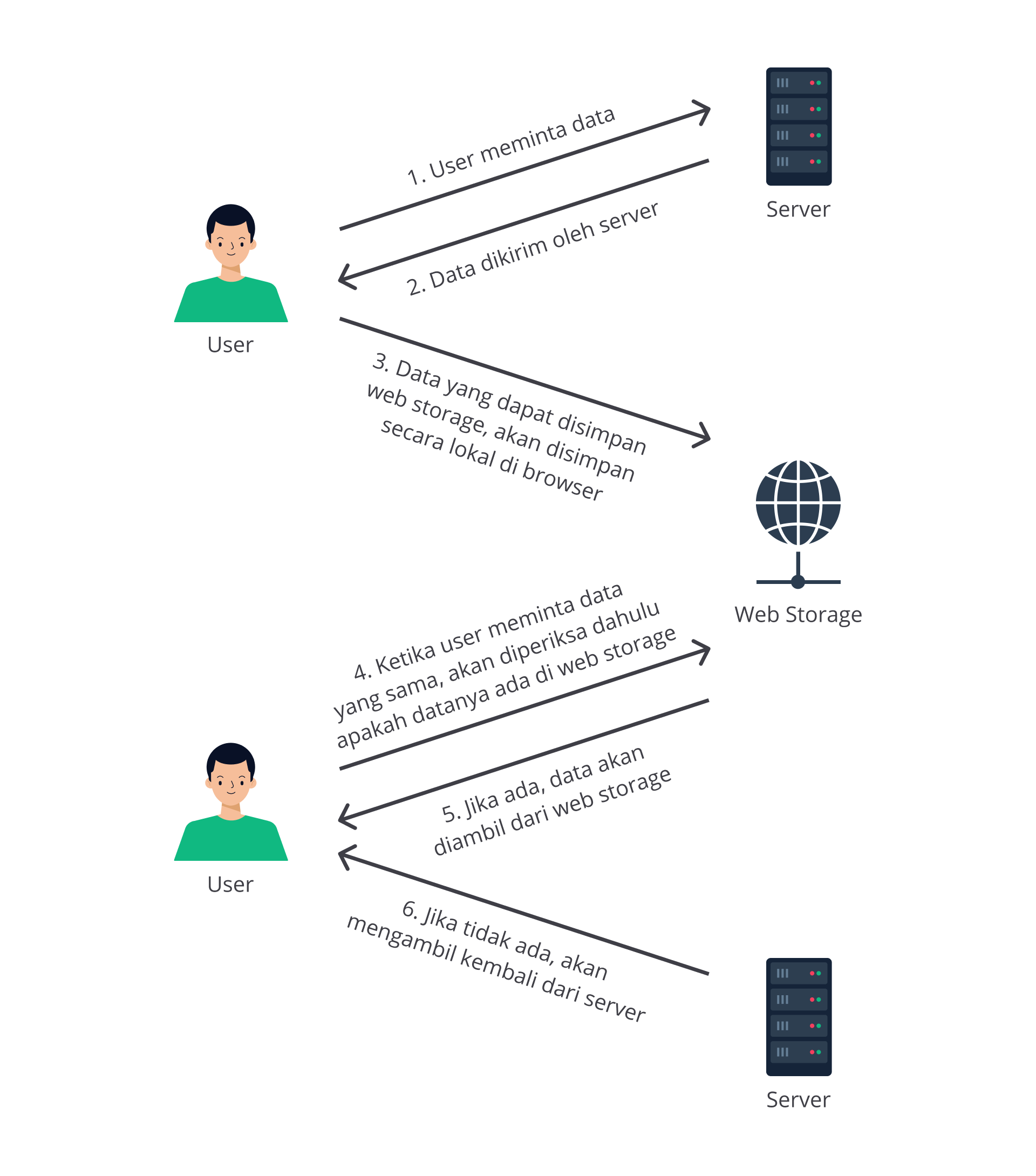
Web storage adalah salah satu Web API (perantara agar kode JavaScript bisa "berkomunikasi" dengan browser) yang dapat menyimpan data secara lokal pada sisi client (disimpan secara lokal pada perangkat kita). Berbeda dengan objek atau array, data yang disimpan pada objek atau array JavaScript bersifat sementara. Ia akan hilang jika terjadi reload atau pergantian URL pada browser. Sedangkan data yang disimpan pada Web Storage akan bertahan lebih lama karena data tersimpan dalam storage browser.

Data yang tersimpan dalam web storage tersedia berdasarkan domain. Artinya, data pada suatu domain web hanya dapat diakses oleh domain itu sendiri. Web storage dapat menampung sebesar 10MB untuk satu domain.

Pada umumnya, kita menganggap jika mengakses sebuah halaman web, data yang dibutuhkan akan dikirim dari database milik server web yang kita akses.



Semua data harus diambil terus menerus dari database ketika kita mengakses halaman web tersebut? Jika proses pengambilan data dilakukan terus menerus dari database setiap kali mengakses halaman web, apakah justru malah tidak efisien?



Karena kita mengunduh data yang sama berulang kali? Di sinilah web storage memiliki kelebihan. Mari simak beberapa contoh kasus di mana web storage lebih cocok digunakan :

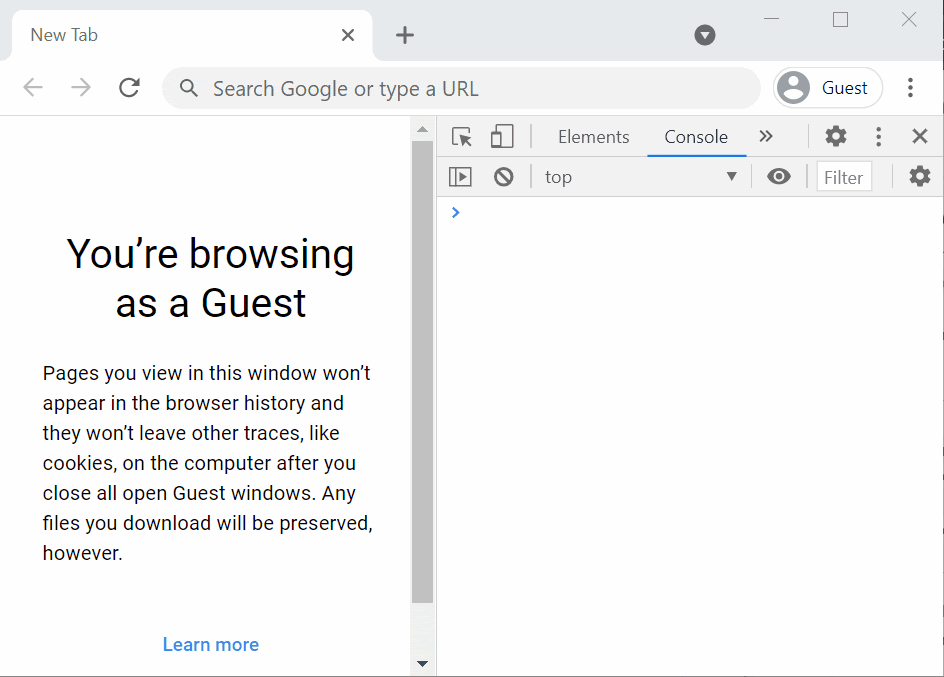
* Menyimpan data dalam bentuk string yang dihasilkan oleh sebuah halaman web agar bisa diakses secara offline.
* Menyimpan pengaturan preferensi pribadi untuk sebuah halaman web contohnya seperti tata letak dan tema warna halaman web.

***Macam-Macam Web Storage dan Cara Mengetahui Dukungan Web Storage.***

Web storage terbagi menjadi 2 yakni local storage dan session storage. Sebelum menggunakan Web Storage baik sessionStorage atau localStorage, kita perlu memastikan browser support terhadap fitur tersebut, dengan cara menjalankan perintah berikut pada console browser :

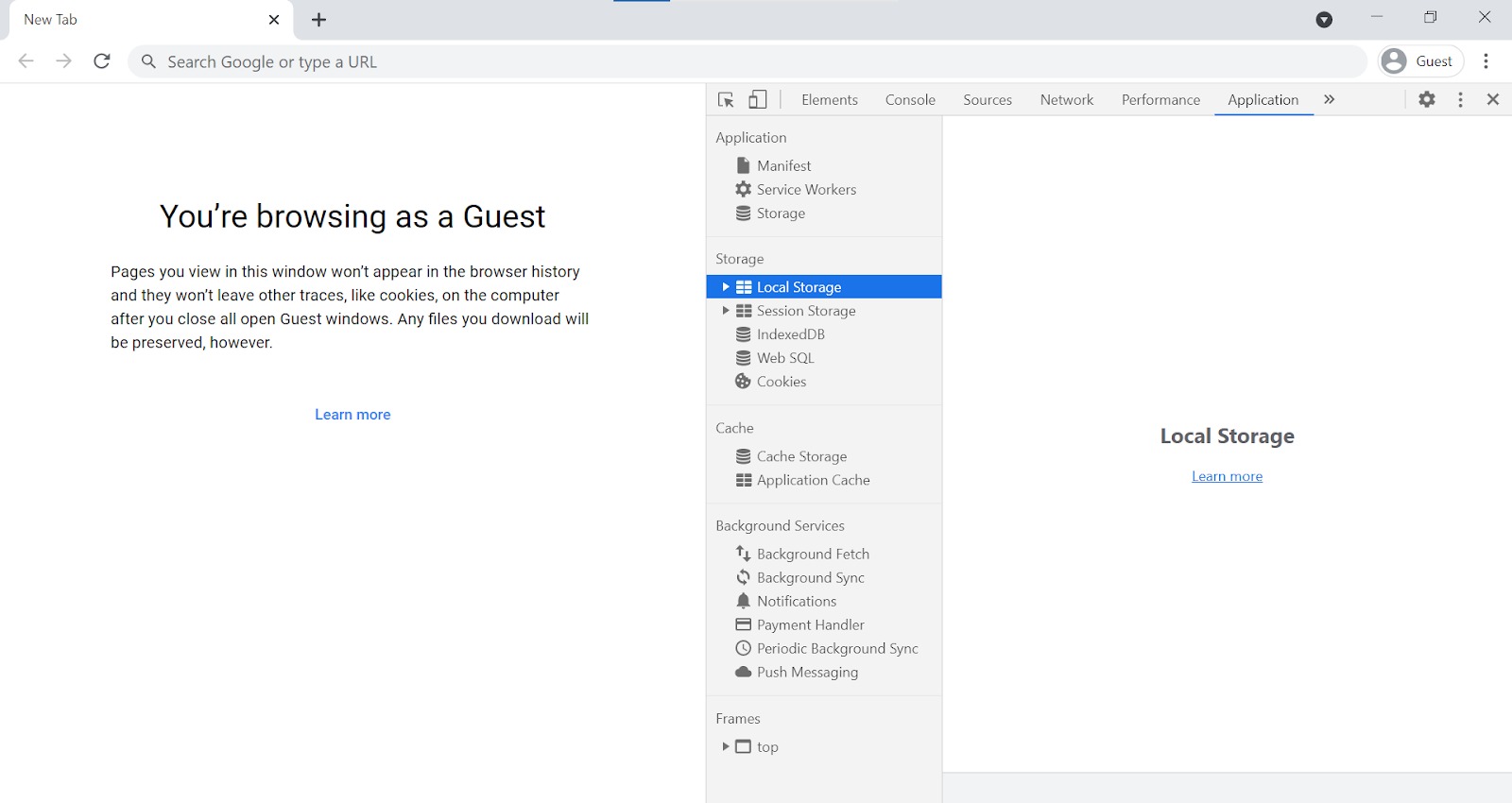
typeof(Storage);

Output yang dihasilkan semestinya tidak bertuliskan "undefined".

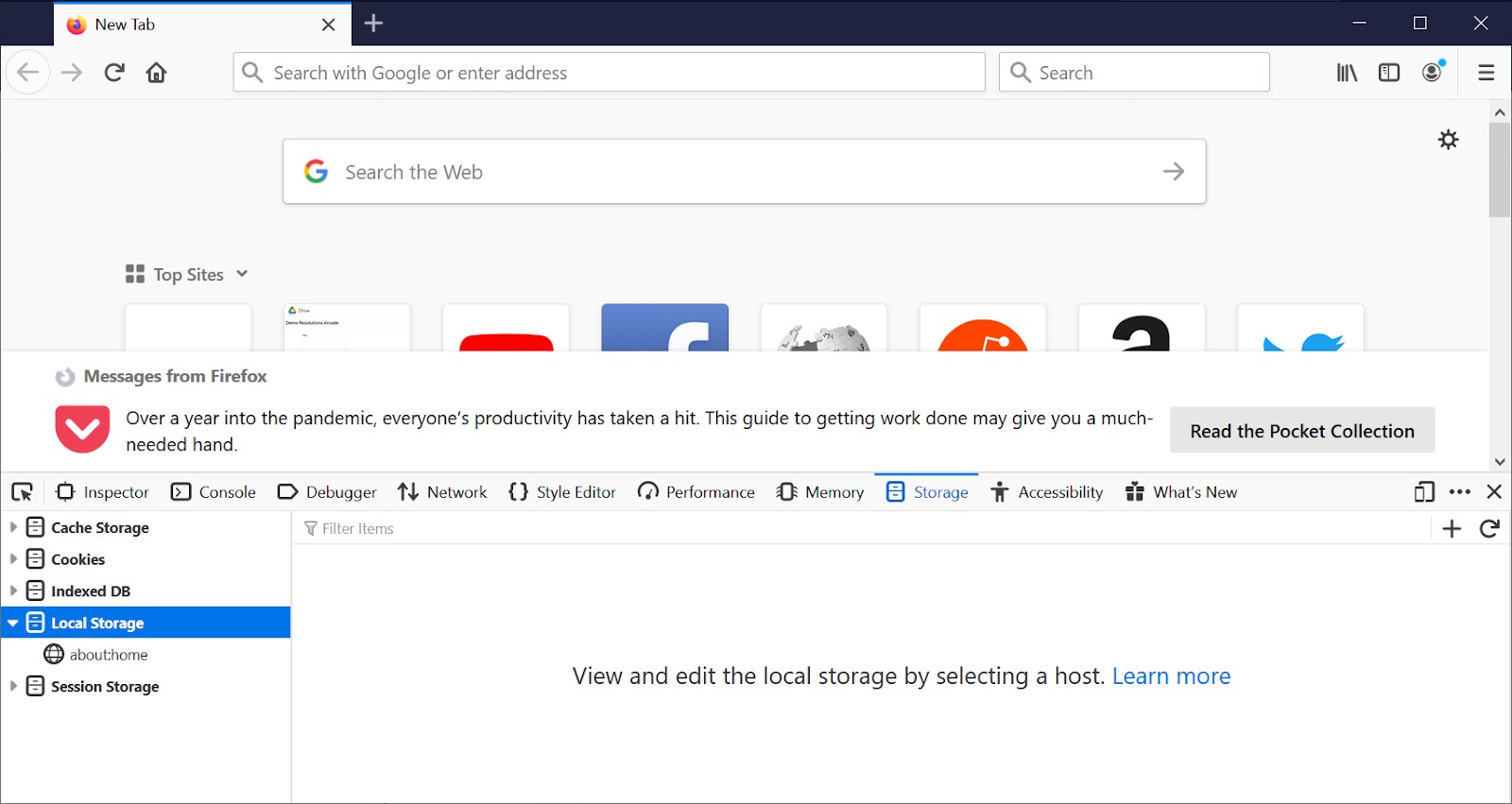


Data yang tersimpan dalam sessionStorage atau localStorage adalah nilai dengan tipe data primitif seperti number, boolean, atau string. Namun, kita juga dapat menyimpan data berbentuk JavaScript objek dengan mengubahnya ke dalam string (JSON) terlebih dahulu, begitu pula sebaliknya ketika kita ingin menggunakan datanya kembali.

Metode yang dapat digunakan untuk menyimpan dan mengakses data pada storage adalah key-value. Di sini key menjadi sebuah kunci untuk mengakses data pada storage. Ibarat sebuah loker, jika Anda ingin menyimpan sebuah benda pasti memerlukan kunci unik untuk loker tersebut. Sama halnya jika Anda ingin mengambil kembali barang tersebut, yakni membukanya dengan kunci yang tepat. Data yang disimpan pada Web Storage dapat kita lihat menggunakan DevTools pada tabApplication (Google Chrome) atau tabStorage (Mozilla Firefox).



Data Local Storage dan Session Storage pada Google Chrome.



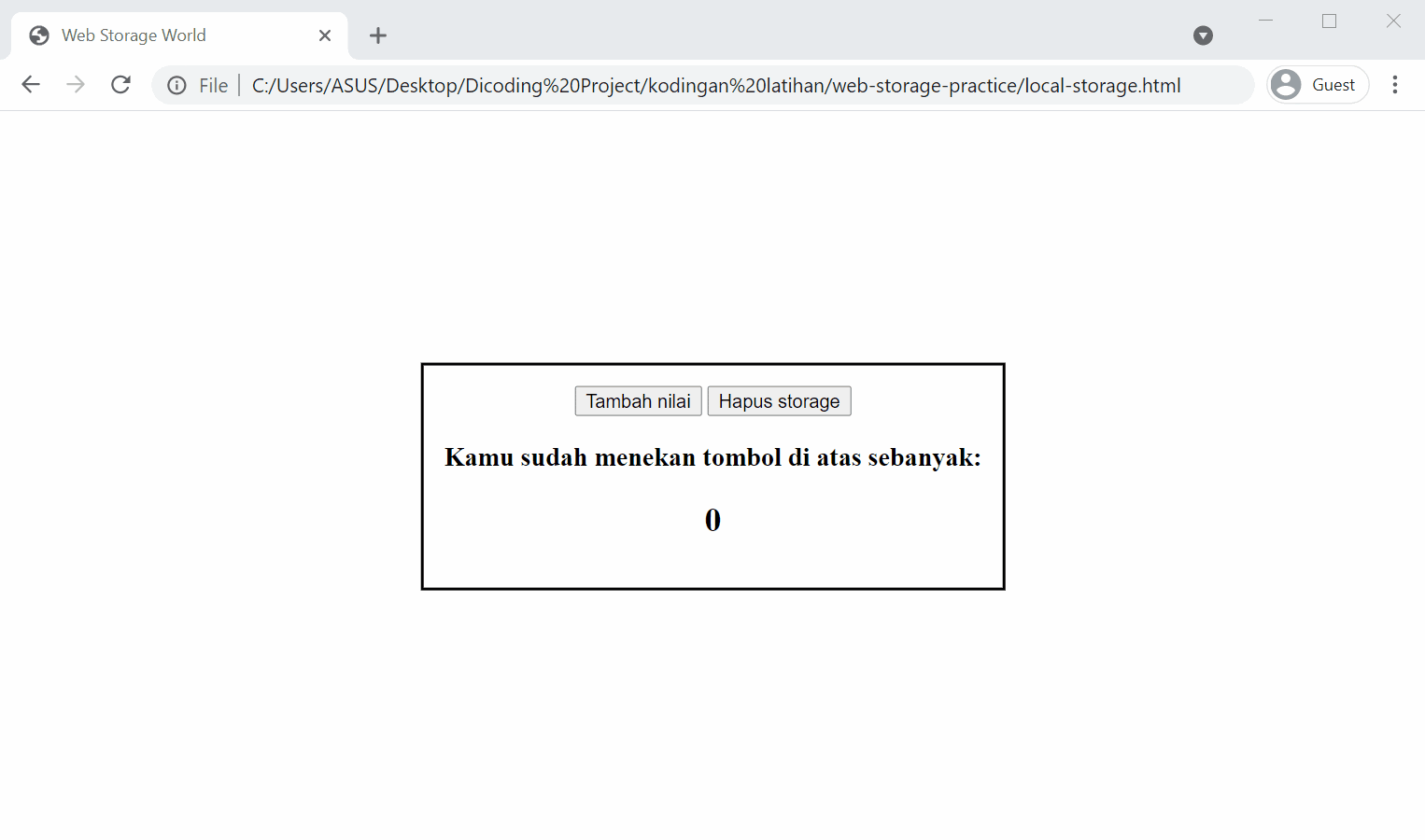
Data Local Storage dan Session Storage pada Mozilla Firefox.

***Local Storage***

Tipe storage yang pertama adalah local storage yang digunakan untuk menyimpan data tanpa ada batasan waktu. Data yang disimpan tidak akan hilang bila browser atau tabs browser ditutup kecuali jika kita menghapusnya.

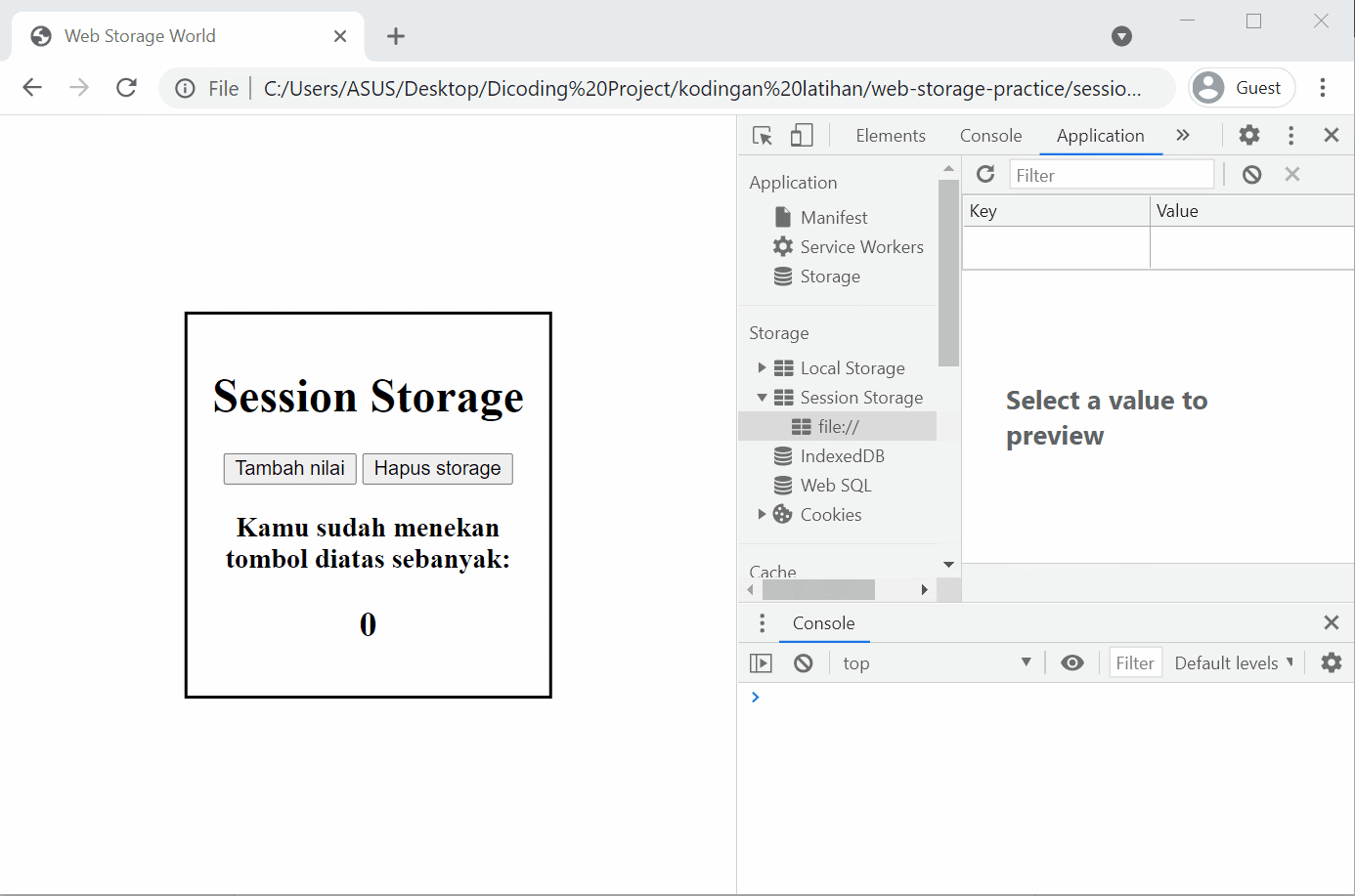
Untuk menggunakan local storage, kita harus mengaksesnya melalui objek yang bernama "localStorage". Kemudian untuk menyimpan datanya, kita harus menggunakan method setItem() dari objek tersebut.

Method setItem() membutuhkan 2 parameter, di mana parameter pertama adalah key dan yang kedua adalah value yang ingin kita simpan. Kemudian untuk mengaksesnya kita memerlukan method getItem() yang membutuhkan 1 parameter yaitu nilai key yang kita gunakan untuk menyimpan data. Seperti yang disebutkan sebelumnya, data pada local storage bisa dihapus dengan method removeItem() yang menerima 1 parameter yakni nilai key yang kita gunakan untuk menyimpan sebuah objek.

***Session Storage***

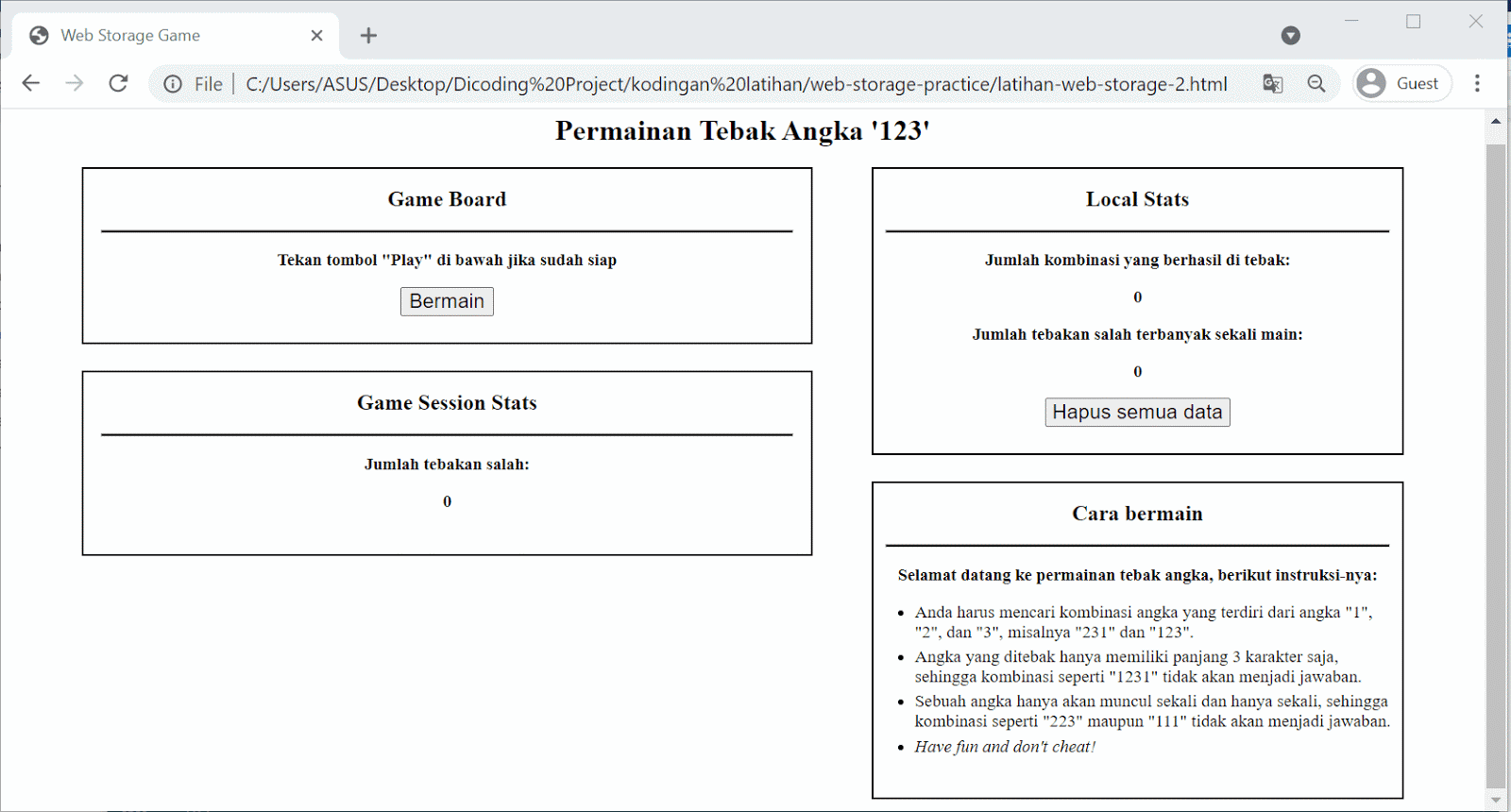
Tipe storage yang kedua adalah Session Storage yang digunakan untuk menyimpan data sementara pada browser. Data akan hilang ketika browser atau tab browser ditutup.

Untuk menerapkan Session Storage, kita dapat menggunakan global objek sessionStorage. Method-method yang tersedia sama dengan milik objek localStorage. Kemudian untuk menyimpan data gunakan method setItem() mengakses data yang sudah dimasukkan dapat menggunakan method getItem() dan untuk menghapus item gunakan method removeItem().



Bisa kita lihat bahwa data yang tersimpan dalam session storage akan bertahan jika terjadi reload/refresh pada browser. Namun, data tersebut akan hilang apabila tab browser atau browser itu sendiri ditutup.

***Implementasi Web Storage - Menambahkan Item Storage***

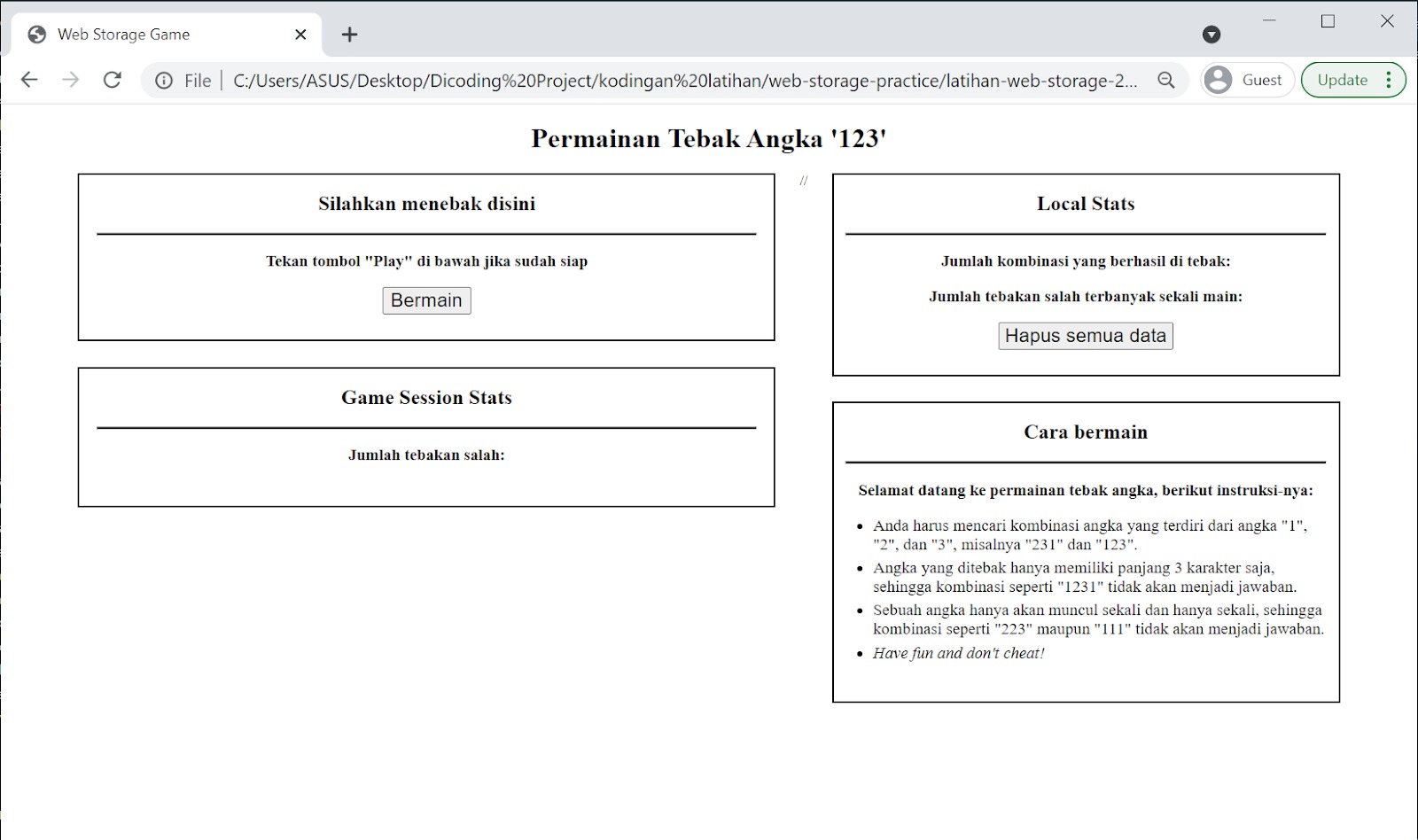


Dengan menggunakan web storage, kita bisa menyimpan berbagai data saat user memainkan permainan tebak angka, seperti berapa banyak user telah menebak, berapa kali user berhasil menebak dengan benar, dan sebagainya.

Di dalam elemen <script>, sudah terdapat beberapa kode antara lain:

* Variabel-variabel yang bisa kita gunakan untuk berinteraksi dengan DOM.
* Sebuah fungsi getAnswer() yang berfungsi untuk menghasilkan kombinasi dari angka "1", "2", dan "3". Kombinasi tersebut akan menjadi jawaban untuk permainan tebak angka.
* Variabel-variabel yang menampung key untuk session storage maupun local storage.

Jika, kita buka halaman web tersebut, layout-nya akan tampil sebagai berikut:



**IKHTISAR**

Anda berada di akhir dari modul Web Storage. Mari kita uraikan materi yang sudah Anda pelajari untuk mempertajam pemahaman.

***Web Storage***

Merupakan salah satu Browser API yang digunakan untuk menyimpan data dengan pola key-value.

Web Storage dapat menampung data maksimal 10MB per domain.

***Fungsi dari Web Storage***

Menyimpan data dalam bentuk string yang dihasilkan oleh halaman web agar bisa diakses secara offline.

Cocok juga untuk menyimpan data konfigurasi preference untuk pengguna web.

***Macam-macam Web Storage***

Local Storage: Digunakan untuk menyimpan data tanpa ada batasan waktu. Data yang disimpan tidak akan hilang bila browser atau tabs browser ditutup kecuali jika kita menghapusnya.

Session Storage: Digunakan untuk menyimpan data sementara pada browser. Data akan hilang ketika browser atau tab browser ditutup.

***Fungsi-fungsi yang ada pada Web Storage:***

setItem: Digunakan untuk menyimpan data pada Web Storage. Fungsi ini membutuhkan dua parameter yakni key (sebagai kunci untuk mendapatkan nilai) dan value (sebagai nilai yang akan disimpan).

getItem: Digunakan untuk mengakses data pada Web Storage. Fungsi ini membutuhkan satu parameter yakni key (sebagai kunci untuk mendapatkan nilai) dan data yang disimpan pada Web Storage akan dikembalikan dalam bentuk string.

Menyimpan dan Mendapatkan Data Kompleks pada Web Storage:

Untuk menyimpan data kompleks seperti objek JavaScript dapat dilakukan dengan mengubah objek menjadi string menggunakan JSON.stringify().

Untuk mendapatkan data kompleks seperti objek JavaScript dari Web Storage dapat dilakukan dengan mengubah string objek menjadi objek menggunakan JSON.parse().

Dengan ringkasan tersebut, diharapkan Anda dapat memahami semua materi yang telah disampaikan. Jika belum, Anda bisa ulas kembali materi yang diberikan pada modul ini. Untuk Anda yang sudah merasa mantap, tinggal selangkah lagi agar Anda bisa lulus dari kelas ini! Di materi selanjutnya, terdapat proyek yang harus Anda kerjakan. Sudah tidak sabar untuk mengetahui apa itu proyeknya? Simak materi selanjutnya yah!