

Dev Web

côté serveur


S3 - R3.01

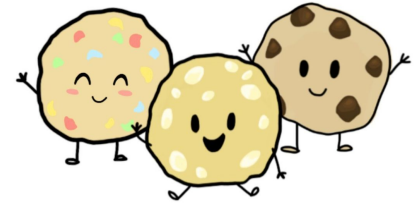


HTTP Cookies

- Les sites web stockent habituellement une petite quantité de données sur les navigateurs.
- Il existe trois principaux types de stockage basé sur le navigateur : le stockage de session, le stockage local, et le stockage de cookies.
- Les cookies enregistrent vos préférences linguistiques, permettant au site web de se souvenir de la langue utilisée lors de vos visites futures.
- Sur les sites e-commerce, les cookies mémorisent les articles ajoutés dans votre panier d'achat, conservant ainsi votre liste d'achats lors de vos visites ultérieures.
- En résumé, un cookie est utilisé pour se souvenir des données de l'utilisateur.

What are
HTTP Cookies?

@Rapid_API 



HTTP Cookies

- Les cookies sont de petites chaînes contenant des paires clé-valeur d'informations envoyées du serveur web au navigateur pour obtenir des informations sur l'utilisateur.
- Le navigateur les sauvegarde localement, permettant ainsi des requêtes ultérieures au serveur pour mettre à jour instantanément le contenu utilisateur sur le site web, en fonction des requêtes précédentes faites par l'utilisateur.
- Un cookie est généré par HTTP et est donc appelé cookie HTTP.

@Rapid_API 



Cookies are small pieces of data sent by a server and stored in the client's browser.

The general purpose of cookies is to identify each user so that websites can adapt their content accordingly.



Although transferred via HTTP protocol which is stateless, cookies allow us to store meaningful state that benefits the functioning of the web.

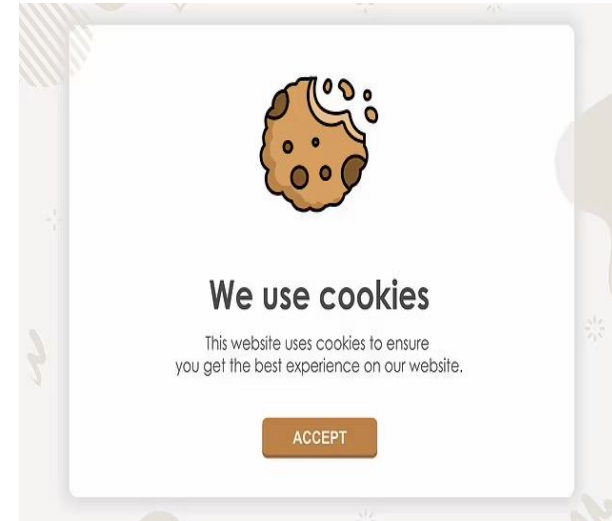
Une brève histoire des cookies

- Le premier cookie HTTP a été créé en 1994 par Lou Montulli, un employé de Netscape Communications, l'entreprise à l'origine du navigateur Netscape. Lou travaillait sur la création d'un magasin en ligne pour une entreprise qui se plaignait que ses serveurs étaient saturés par le stockage des données des paniers d'achat des utilisateurs.
- Pour résoudre ce problème, Lou a eu l'idée de stocker localement le contenu des paniers d'achat. Il a donc conçu un moyen de sauvegarder les informations du panier d'achat sur l'ordinateur de l'utilisateur afin d'économiser l'espace serveur. Il s'est inspiré du concept de "magic cookie", un jeton informatique utilisé pour identifier un utilisateur lors de la connexion à un système.
- Lou a adapté ce concept et l'a implémenté dans un navigateur web. En 1994, le navigateur Netscape a intégré les cookies, suivi par Internet Explorer en 1995, marquant ainsi la naissance des cookies HTTP.



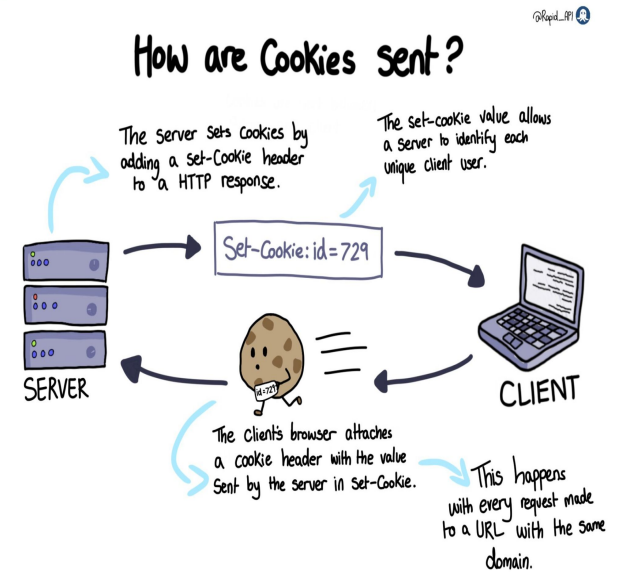
Comment fonctionnent les cookies - 1 -

- Lorsqu'un utilisateur visite pour la première fois un site web qui utilise des cookies, le navigateur l'informe de l'utilisation de cookies et demande à l'utilisateur d'accepter leur sauvegarde sur son ordinateur.
- Habituellement, lorsqu'une requête est faite par l'utilisateur, **le serveur répond en renvoyant un cookie**, parmi d'autres éléments.
- Ce cookie est stocké dans le navigateur de l'utilisateur. Lorsque l'utilisateur visite à nouveau le site ou envoie une autre requête, **cette requête est renvoyée avec les cookies**.
- Le cookie contient certaines informations sur l'utilisateur que le serveur peut utiliser pour prendre des décisions sur les requêtes subséquentes.




Comment fonctionnent les cookies - 2 -

- Un exemple parfait est l'accès à Facebook depuis un navigateur. Pour accéder à votre compte Facebook, vous devez vous connecter avec les identifiants corrects.
- Lors de la première connexion, le serveur vérifie vos identifiants et envoie le contenu de votre compte Facebook ainsi que des cookies à votre navigateur.
- Ces cookies, stockés sur votre ordinateur, sont soumis au serveur à chaque nouvelle requête sur ce site. Un cookie sera enregistré avec un identifiant unique pour cet utilisateur.
- Lorsque vous revisitez Facebook, la requête que vous faites, le cookie enregistré et le serveur suivent votre session de connexion, se souviennent de qui vous êtes et vous maintiennent connecté.



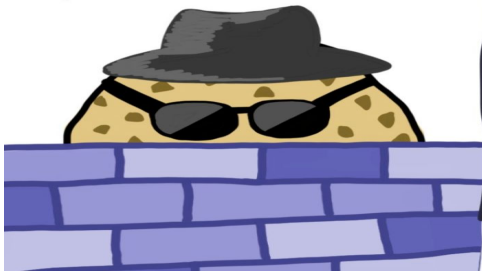
Types et finalités des cookies

@Rapid_API 

Cookies have three Primary Purposes:

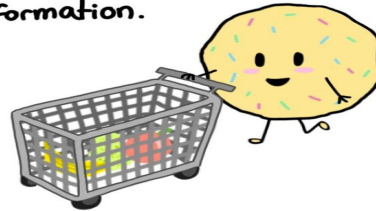
Tracking

Tracking builds statistics about the user, and this data can be used for ad personalization.



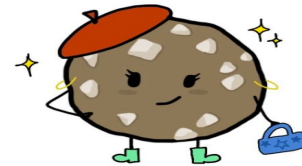
Session management

Cookies store data from sites so that when you return your changes are kept. For example - items you put into a shopping cart, or login information.



Personalization

Cookies can also retain data on user preferences, such as language, themes, location, and layout.



Types et finalités des cookies

- **Cookies de session** : stockent les informations de l'utilisateur pour une courte période. Ils sont supprimés de l'ordinateur de l'utilisateur à la fin de la session actuelle.
- **Cookies persistants** : un cookie persistant n'a pas de date d'expiration. Il est conservé aussi longtemps que l'administrateur du serveur web le définit.
- **Cookies sécurisés** : utilisés par les sites web cryptés pour offrir une protection contre les éventuelles menaces de piratage.
- **Cookies tiers** : utilisés par les sites web qui affichent des publicités sur leurs pages ou suivent le trafic web. Ils permettent à des parties externes de décider des types de publicités à afficher en fonction des préférences antérieures de l'utilisateur.

Cookie Lifetime

@Rapid_API

Cookies have different lifetimes depending on whether they are:

Permanent Cookies

or

Session Cookies

This type uses the Expires attribute to set a specified date that the cookie will be deleted.

```
Set-Cookie:  
id=B125;  
Expires=Sat, 2 Jul 2022, 08:00:00 GMT;
```

You may also use the Max-Age attribute to do this.



Session Cookies are deleted when the current browser session ends. If the browser restores when restarting, some cookies can remain permanently.

Sessions VS Cookies

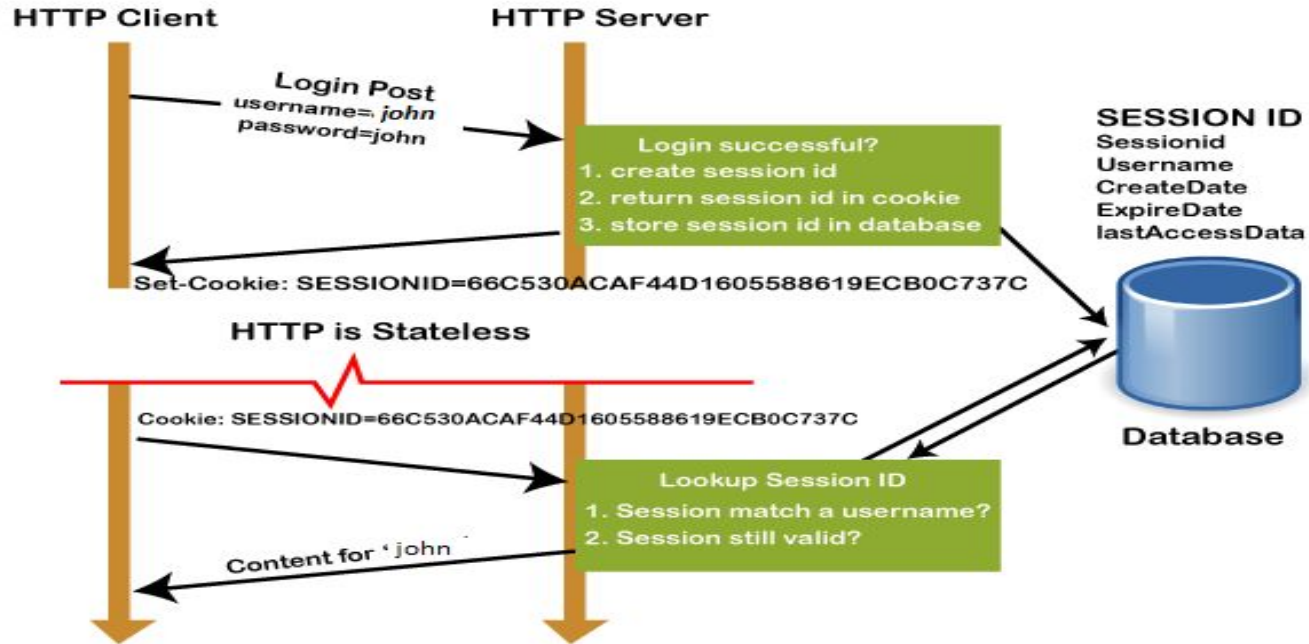
- La principale différence entre les sessions et les cookies est que **les sessions résident côté serveur (le serveur web)**, tandis que **les cookies se trouvent côté client (dans le navigateur de l'utilisateur)**. **Les sessions contiennent des informations sensibles** telles que les noms d'utilisateur et les mots de passe, c'est pourquoi elles sont stockées sur le serveur. Elles peuvent être utilisées pour identifier et valider l'utilisateur qui effectue une requête.
- Comme expliqué, **les cookies sont stockés dans le navigateur et ne peuvent pas contenir d'informations sensibles**. Ils sont typiquement utilisés pour enregistrer les préférences d'un utilisateur.



Sessions/Cookies en NodeJS - Express

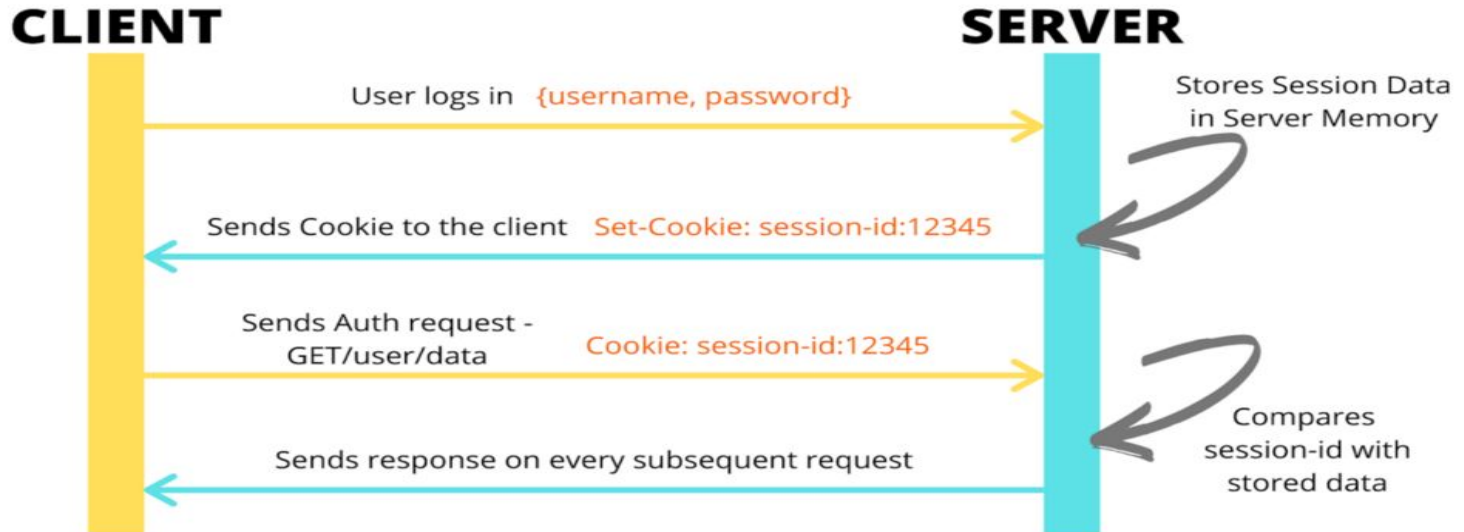
- Il existe deux principaux modules de gestion des sessions par cookies en middleware : **express-session** et **cookie-session**.
- La principale différence entre ces deux modules réside dans la manière dont ils sauvegardent les données de session des cookies. Le middleware **express-session stocke les données de session sur le serveur** ; il enregistre uniquement l'ID de la session dans le cookie lui-même, et non les données de la session. Par défaut, il utilise un stockage en mémoire et n'est pas conçu pour un environnement de production. En production, vous devrez configurer un système de stockage de sessions évolutif ;
- En revanche, le middleware **cookie-session** met en œuvre un stockage basé sur les cookies : il sérialise toute la session dans le cookie, plutôt que juste une clé de session. Utilisez-le uniquement lorsque les données de session sont relativement petites et facilement codables en tant que valeurs primitives (plutôt qu'en objets). Bien que les navigateurs soient censés supporter au moins 4096 octets par cookie, pour éviter de dépasser cette limite, il est recommandé de ne pas dépasser une taille de 4093 octets par domaine. De plus, soyez conscient que les données du cookie seront visibles par le client, donc si vous avez une raison de les garder sécurisées ou obscures, alors **express-session** peut être un meilleur choix.

Authentication basée sur la session



Authentication basée sur la session

Session Based Authentication



Cookie: SameSite

