

## Session:

A session is server side information intended to exist only throughout the visitor's interaction with the website. If you don't need any permanent data, sessions are usually the way to go. In Laravel, Session is a parameter passing mechanism which enables us to store data across multiple requests.

- **File:** Session Data is stored in an encrypted file located at `storage/framework/sessions`
- **Cookies:** Session data is stored in secure
- **DB:** Session data is stored in Application DB

Laravel session configuration file is located at `app/config/session.php`

## Why Session:

- Little easier to use
- It can be as large as needed in comparison with the cookies
- Tighter security
- Ability to control data
- Ability to control when it expires

## Cookies:

Cookies are text files with small pieces of data like a username and password that are used to identify your computer as you use a computer network. For example, user's username and password are saved as cookies on website so that he need not to log in on each visit.

## Common Usage of Cookies:

- Authentication
- Storage of site preferences
- Shopping Cart item

[Note: Don't use cookies to store sensitive data]

## Session vs Cookie:

In PHP, visitor information designated to be used across the site can be stored in either session or cookies. They do much the same things.

Session	Cookies
<ul style="list-style-type: none"><li>• Information stored in the session is stored at web server</li></ul>	<ul style="list-style-type: none"><li>• Visitor's information stored in the cookie is stored on visitor's browser</li></ul>
<ul style="list-style-type: none"><li>• Sessions can be disabled or edited by the visitor</li></ul>	<ul style="list-style-type: none"><li>• Sessions can not be disabled or edited by the visitor</li></ul>

[Note: In advance, we have to make a combination of sessions and cookies to make our website as like as we want]

## **Tasks using Laravel Session {A-Z}:**

- ❖ Login with session
- ❖ Session login with timeout (set lifetime)
- ❖ Session login with browser close response (set expires\_on\_close)
- ❖ Session login with validation by OTP
- ❖ Session login with OTP expire time (responsive remaining timer)
- ❖ Session login with a second and last chance to resend OTP if user fails first time

[Tools & Technologies:: VS CODE EDITOR, HTML, Laravel(PHP), JS]