ACADEMIND

What is “Authentication”?

In many websites, certain “areas” (pages) should only be accessible by authenticated (= logged in) users

- Personal profile on a social network site
- Your shopping cart and order history in an online shop
- The administration area of your own blog website

User **Sign Up** *(create account with email + password)* → User **Login** *(enter email + password)* → User **Authentication** *(grant access to protected pages)*

ACADEMIND

Understanding Password Hashing

To render security-relevant data (e.g. a password) useless in case of a data breach, you should hash it

Hashing = converting a string (e.g. the password) to a **non-decodable**, different string

Securely hashed values can’t be reverted, decoded or transformed back into the original value

ACADEMIND

We Need To Track The User Auth Status

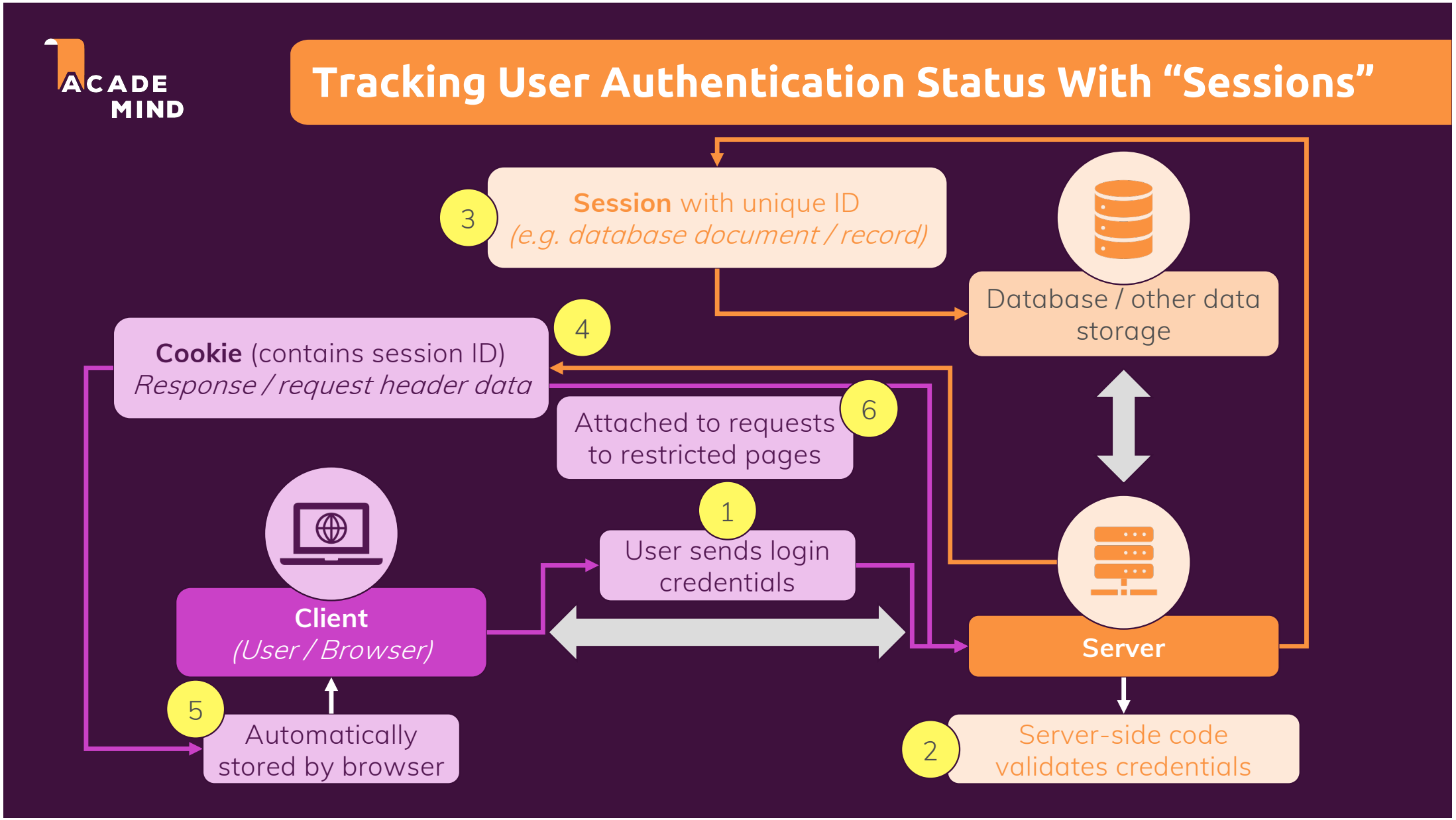
To the web server (backend code), every incoming request is similar

Just by looking at a default request, the server-side code can't find out whether a user should be granted access or not

An "entry ticket" must be saved on the server and handed out to the visitor

ACADEMIND

Tracking User Authentication Status With "Sessions"



ACADEMIND

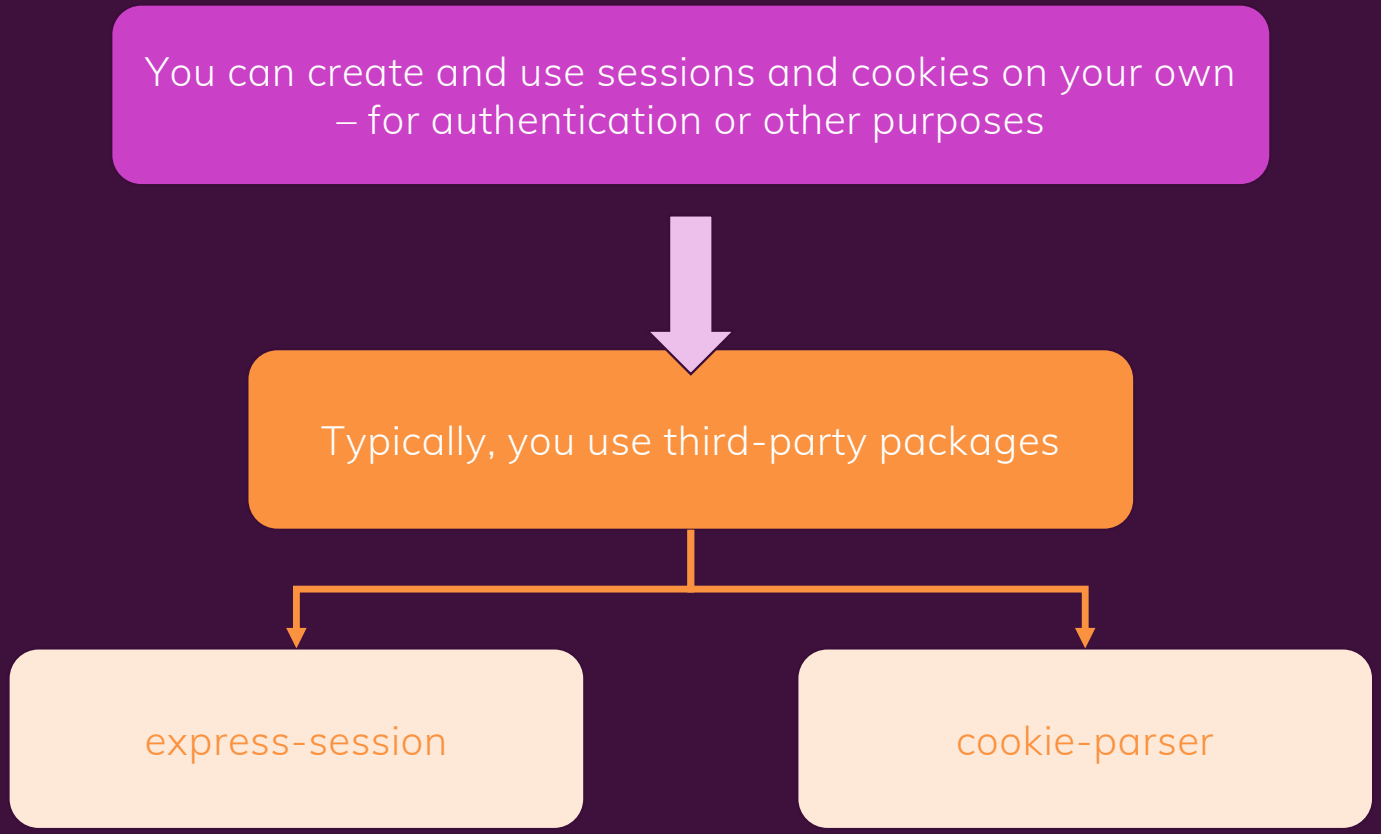
A Closer Look At Sessions & Cookies

Sessions & Cookies are NOT exclusively used for authentication!

- Unauthenticated users can also be assigned sessions
- The data stored in a session (e.g. “isAuthenticated: true”) matters
- Cookies are also used for user tracking etc.
- Cookies can be combined with sessions – but don’t have to

ACADEMIND

Working With Sessions & Cookies



ACADEMIND

Authentication vs Authorization

Authentication	Authorization
Signup + login with credentials	Even authenticated users may not be allowed to access everything on a website
Authenticated (= logged in) user may then access restricted resources	E.g. not all authenticated users may access your online shop order history