



# Sessioni



# HTTP

- HTTP è un protocollo *stateless*, ossia ogni coppia *client-request/server-response* è indipendente dalle altre.
- Tuttavia, per ovviare a questo limite, sono disponibili le seguenti tecniche:
  - Cookie
  - Sessioni
  - File
  - Database



## Limiti dei cookie

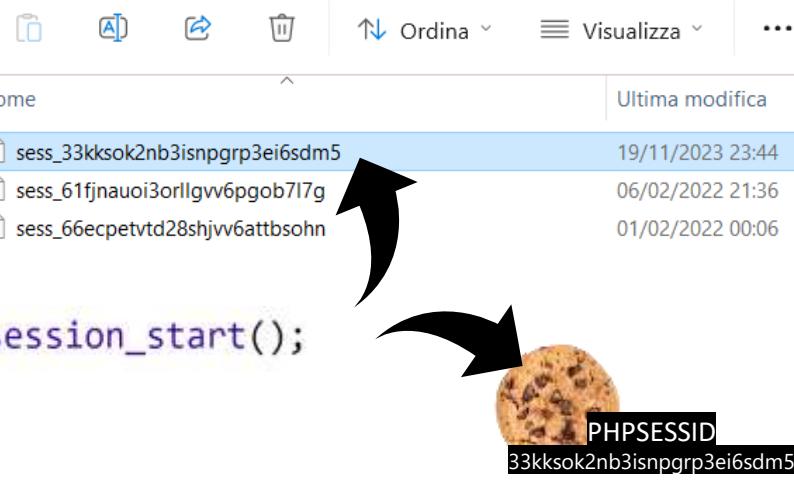
- Le informazioni contenute nei cookie possono essere facilmente accessibili lato client e quindi anche modificabili;
- Possono contenere una quantità limitata di dati (solitamente 4KB).





## Sessioni: idea di base

- A differenza dei cookie, con le sessioni i dati da memorizzare sono salvati lato server.
- Il server crea una **sessione** per ciascun client, in particolare crea:
  - un file di testo identificato da una stringa univoca detta PHPSESSID (ad es. 33kksok2nb3isnpgrp3ei6sdm5), al cui interno verranno memorizzati i dati
  - Un cookie contenente lo stesso PHPSESSID



# Esempio



GET pag1.php?name=Andrea HTTP/1.1

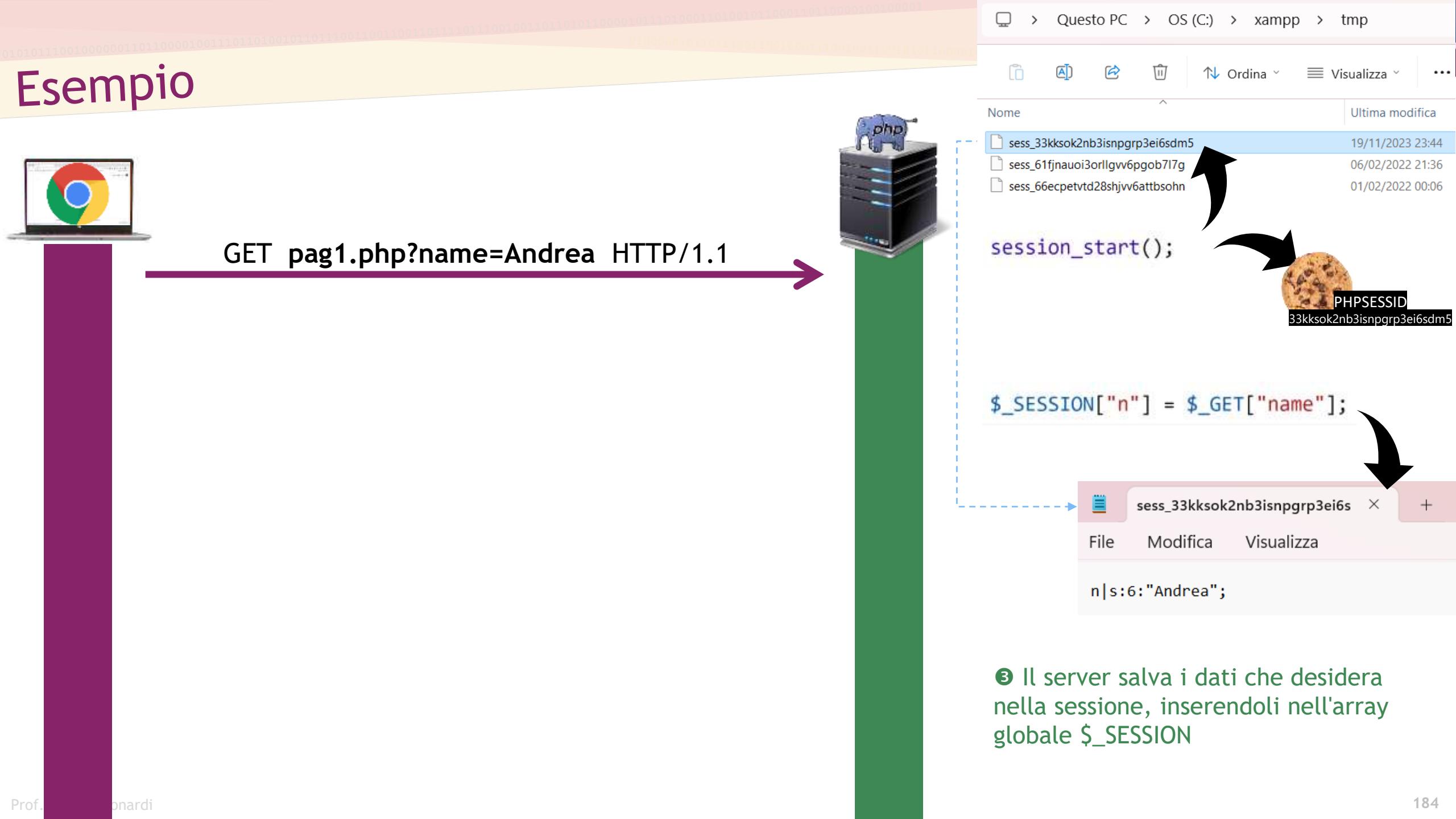
① Il client contatta per la prima volta il server



`session_start();`

② Il server avvia una nuova sessione:

- Genera un PHPSESSID
- Crea un file specifico per questa nuova sessione
- Crea un cookie di sessione

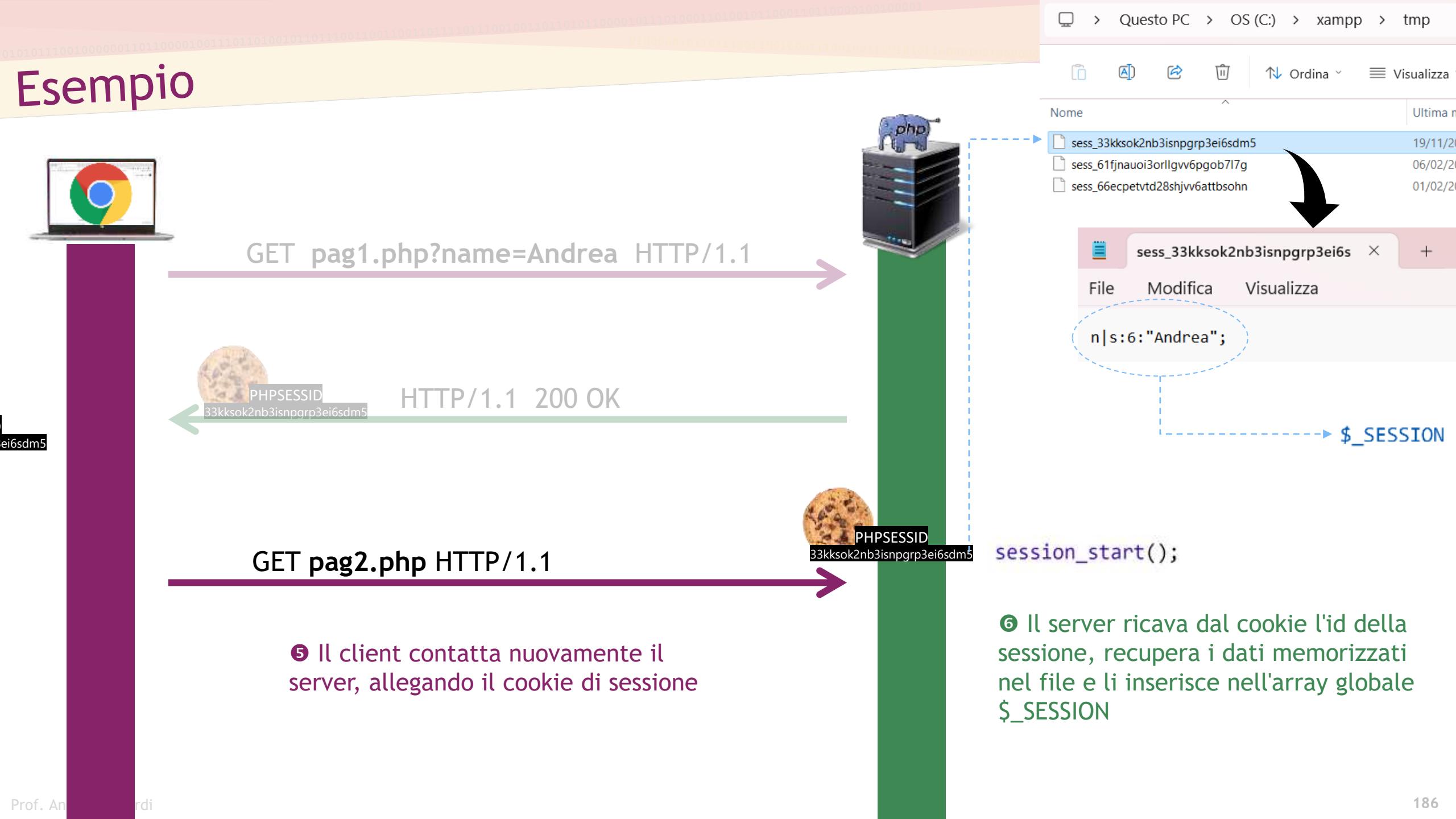


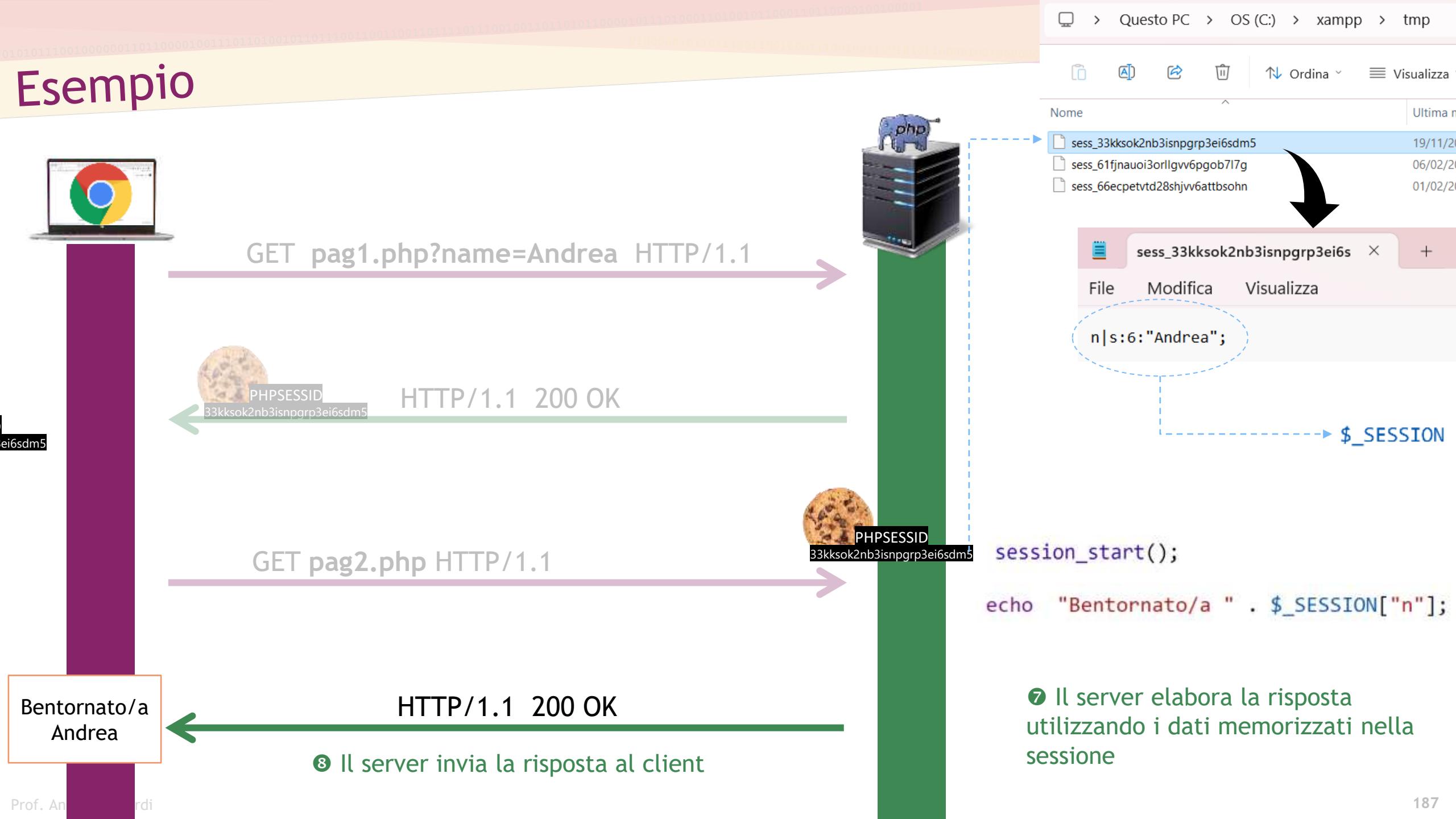
- ③ Il server salva i dati che desidera nella sessione, inserendoli nell'array globale `$_SESSION`



# Esempio









# Sessioni in PHP

- **`$_SESSION`** è un array associativo globale contenente le variabili di sessione
  - È disponibile solo dopo aver invocato la funzione `session_start()`
  - Per modificare una variabile è sufficiente sovrascriverla
  - Per rimuovere una variabile si può utilizzare la funzione  
`unset($_SESSION["nomeVar"]);`



# Sessioni in PHP

- **session\_start()** è la funzione che crea una nuova sessione oppure ripristina la sessione creata in precedenza
- Deve essere presente in tutte le pagine PHP che utilizzano i dati della sessione
- Va posizionata all'inizio della pagina, prima di qualsiasi output
- Se il cookie inviato dal client contiene un id di sessione riconosciuto dal server, i dati presenti nel file di sessione vengono inseriti nell'array **\$\_SESSION**.



# Sessioni in PHP

- Una sessione termina nel momento in cui l'utente chiude il browser.
- `session_unset()`, rimuove tutte le variabili di sessione
- `session_destroy()`, per eliminare la sessione corrente



# Cookie VS Sessioni

Caratteristica	Cookie	Sessione (server-side)
Memorizzazione	Lato client (nel browser)	Lato server
Persistenza	Può essere <b>persistente</b> (con scadenza) o <b>di sessione</b> (finché il browser è aperto)	Temporanea, scade alla chiusura del browser o dopo un timeout
Visibilità	Accessibile da <b>client e server</b>	Accessibile <b>solo dal server</b>
Sicurezza	Meno sicuro (può essere modificato dal client)	Più sicuro (l'utente non può accedere ai dati direttamente)
Spazio disponibile	Limitato (~4KB per cookie)	Più spazio disponibile (dipende dalla RAM/server)
Identificatore	Contiene direttamente i dati o un identificatore	Contiene solo un identificatore (di solito salvato in un cookie)
Dipendenza	Funziona anche senza sessioni	Spesso <b>usa un cookie</b> per salvare l'ID di sessione
Gestione dati	I dati sono <b>memorizzati nel cookie stesso</b>	I dati sono <b>memorizzati nel server</b>
Uso tipico	Salvataggio preferenze, tracciamento, login automatico	Autenticazione utente, carrelli acquisti, dati temporanei sensibili
Durata	Può durare <b>giorni, mesi o anni</b>	Di solito <b>scade velocemente</b>



# Approfondimento

- Come è possibile rimanere connessi a uno stesso sito web per più giorni?

