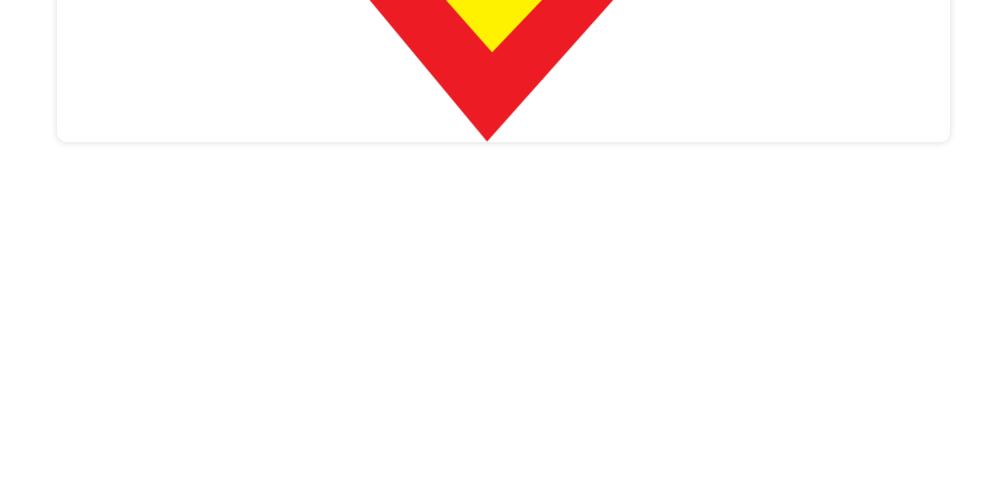
LOGIN / LOGOUT

\$_SESSION

SUPERGLOBALS





SUPERGLOBALS

- \$GLOBALS
- \$_SERVER
- \$_GET
- \$_POST
- \$_FILES
- \$_COOKIE
- \$_SESSION
- \$_REQUEST
- \$_ENV

\$_GET för att få URL-query

```
https://mysite.com?name=jesper&ok=whatev
```

```
var_dump($_GET) // { name: jesper, ok: whatev }
```

\$_POST för att hämta data från forms

```
<form action="post.php" method="POST">
    <input type="text" name="name " value="jesper">
        <input type="text" name="ok" value="whatev">
        <input type="submit">
        </form>
```

```
var_dump($_POST) // { name: jesper, ok: whatev }
```

form action, form method och name är viktiga

\$_SESSION

I sessionsvariabeln kan vi lagra information som ska vara under en längre tid

En session håller data när du går till en annan sida.

Lagras på servern och i browsern men inte i databasen.

```
$_SESSION['name'] = "Jesper";
```

Varje gång du ska köra en session måste du se till så att sessionen är startad

```
session_start();
session_destroy();
```

En session kan t.ex. vara en inloggning

Vi ska inte starta sessionen om den redan är startad.

```
if (session_status() == PHP_SESSION_NONE) {
    session_start();
}
```

Lägg session_start() längst upp i dokumentet

Vill vi ta bort en variabel i session

unset(\$_SESSION["username"])

LOGIN/LOGOUT

\$_SESSION används främst i kombination med login / logout men man kan lagra vad som helst i arrayen

Vi vet hur vi kan lagra en användare/vad som helst i SQL

```
INSERT INTO users (username, password)
VALUES (zero_cool, strongpass2000)
```

Vad är felet? 🥮

```
INSER INTO users (username, password)
VALUES (zero_cool, strongpass2000)
```

Tips: säkerhet

Vi måste kryptera våra lösenord

Det är inbyggt 🥯



\$hash = password_hash("strongpass2000", PASSWORD DEFAULT)

\$hash innehåller en hashad version av vårt lösenord

Men vi måste kunna avkryptera vårt lösenord

Det är inbyggt 🎃

```
$hash = password_hash("strongpass2000", PASSWORD_DEFAULT)

if (password_verify('strongpass2000', $hash)) {
    echo "Yes! It's YOU!";
} else {
    echo "Liar";
}
```

password_verify returnerar true eller false

baserat på om hashen stämmer överens med lösenordet

Det jobbiga är flödet. Hur ska vi skicka informationen?

Var ska användaren sparas?

Var ska felmeddelanden sparas?