**Osto- ja myyntikanava**

**Rekisteröinti**

* Olemme luoneet osto- ja myyntikanavaamme varten jo tiedoston nimeltä **rekisterointi.html** käyttäjän rekisteröitymistä varten
* Olemme luoneet osto- ja myyntikanavaamme varten jo tiedoston nimeltä **kantayhteys.php** tietokantaan yhdistämistä varten.

Luodaan tämän jälkeen **kayttajatunnistus.php** niminen PHP-tiedosto, jolla hallitaan käyttäjään

liittyviä tietoja.

Avataan koodia hieman, jota tarvitaan, että saadaan käyttäjä rekisteröityä osto- ja myyntikanavalle. Koodi muodostuu seuraavasti.

1. Ensimmäinen komento on sessiota eli istuntoa ja määritellään se alkaneeksi *session\_start()-* funktiolla.
2. Muodostetaan yhteys tietokantaan aiemmin luotua kantayhteys.php sivua käyttäen, jonka koodi liitetään osaksi nykyistä sivua *include-* toimintoa käyttäen.

include("kantayhteys.php");

1. Määritellään $sivu- muuttuja, joka käytännössä lukee, mistä lomakkeesta on lähetetty tietoja sivulle.

$sivu = $\_POST['lomaketunnistin'];

1. Määritellään yleiset muuttujat $kayttaja\_tunnus sekä $kayttaja\_salasana, jotka haetaan

tietoja lähettäneestä lomakkeesta. Käytä muuttujien hakemisessa **Mysqli\_real\_escape\_string -f**unktiota (suojaa tietokantaa verkkohyökkäyksiltä). Tämä on yksi tapa luoda suojaus.

Kerro miten funktiolla hoidetaan tietojen suojaus?

1. Seuraavana tulee if- lause, joka tunnistaa, mistä lomakkeesta tietoja on juuri lähetetty. Rekisteröitymislomakkeessa olevan piilotetun lomaketunnistin-kentän arvoksi on

määritelty 0, joten ohjelma suorittaa lomaketta lähetettäessä aina tämän lohkon koodista.

if($sivu == 0) { suoritettava koodi }

1. Seuraavaksi koodiin luodaan if-else- lauseella suojaus tyhjiä kenttiä vastaan. Mikäli jokin lomakkeen

kentistä jää tyhjäksi ja varmistusluku ei ole “kuusi”, ohjelman pitää antaa virhepalaute.

”Jätit tietoja täyttämättä. Ole hyvä ja <a href='rekisterointi.html'>täytä lomake uudestaan</a>.";

Mikäli kaikki lomakkeen kentät olivat täytettyjä ja varmistusluku “kuusi”,

koodi ajetaan ja ohjelman pitää antaa palaute.

"Rekisteröinti onnistui! <a href='kirjautuminen.html'>Kirjaudu sisälle</a> palveluun.";

Tämän jälkeen tutkitaan, onko lomakkeeseen syötettyä tunnusta jo olemassa vertailemalla

tietokannassa olevia tietoja lomakkeeseen syötettyihin tietoihin. Mikäli tunnus on jo

käytössä, ohjelma antaa palautteen.

1. Määritellään muuttuja nimellä $query, joka sisältää SQL-hakukyselyn. Tässä hakukyselyllä valitaan kaikki tiedot ‘kayttajat’-taulusta siltä riviltä, jossa kayttaja\_tunnuskentän arvo on sama kuin lomakkeesta kerätty käyttäjätunnus.

$query = mysqli\_query($dbconnect, "SELECT \* FROM kayttajat WHERE kayttaja\_tunnus = '$kayttaja\_tunnus'");

1. Kysely aktivoituu vasta edellistä kohtaa seuraavassa if- lauseessa, joka tutkii **mysqli\_num\_rows-** funktion avulla, montako vastinetta kyselylle löytyy. Mikäli vastine löytyy (!== 0), ohjelma kertoo käyttäjälle tunnuksen olevan jo käytössä seuraavalla tavalla.

"Tunnus on jo käytössä! <a href='rekisterointi.html'>Kokeile uudestaan</a>.";

if(mysqli\_num\_rows($query) !== 0) {

echo "Tunnus on jo käytössä! <a href='rekisterointi.html'>Kokeile uudestaan</a>.";

}

1. Mikäli tunnus on puolestaan uusi, siirrytään syöttämään tietoja tietokantaan. $query-muuttuja

saa uuden tehtävän. Tiedot lähetetään tietokantaan syötettyjen rivien mukaisessa järjestyksessä:

1. kayttaja\_id: ‘’ eli ei mitään, tietokanta generoi itse juoksevan numeron tietueelle

2. kayttaja\_taso: user eli alempi käyttäjätaso

3. kayttaja\_tunnus: lomakkeeseen syötetty tunnustieto

4. kayttaja\_salasana: lomakkeeseen syötetty salasanatieto

5. kayttaja\_sahkoposti: lomakkeeseen syötetty sähköpostitieto

else {

$query = mysqli\_query($dbconnect,”INSERT INTO kayttajat (kayttaja\_id, xxxxxxxxxxxxxx) VALUE(‘user’, xxxxxxxxxx ) “

1. Huomaa, että toisin kuin edellisessä if-lauseessa, else-lauseen osalta kysely aktivoidaan

lopuksi **mysqli\_query($query);** lauseella– tätä ennen ei tapahdu mitään.

Jos rekisteröinti onnistui, niin tulostetaan:

”Rekisteröinti onnistui! <a href=’kirjautuminen.html’>Kirjaudu sisälle</a> palveluun.”;

1. Sivun lopussa yhteys tietokantaan suljetaan. Anna komento, jolla yhteys tietokantaan suljetaan.
2. Testataan tämän jälkeen rekisteröintilomakkeen toimivuutta. Avataan selaimeen rekisterointi.html tiedosto ja syötetään siihen tiedot.

Vaikka ohjelma kertoisikin rekisteröinnin onnistuneen, kannattaa silti tarkistaa tietokannasta phpMyAdmin työkalun avulla, onko syötetty tieto mennyt perille (valitse tietokanta → kayttajat-taulu

Myös virhepalautteen toimivuus kannattaa testata syöttämällä kaksi kertaa sama tunnus, jättämällä kenttiä tyhjiksi sekä syöttämällä väärät merkit varmistuskenttään.

Ellei ohjelma ei suostu toimimaan, tarkista koodisi tarkasti. Koodin toimivuus on monesti hyvin pienestä kiinni.