



PHP-TILASTOINTIOHJELMISTO

Miikka Tikka

Opinnäytetyö
Toukokuu 2012
Tietotekniikan
koulutusohjelma
Sulautetut järjestelmät ja
elektroniikka

TAMPEREEN AMMATTIKORKEAKOULU
Tampere University of Applied Sciences

TIIVISTELMÄ

Tampereen ammattikorkeakoulu
Tietotekniikan koulutusohjelma
Sulautetut järjestelmät ja elektroniikka

TIKKA MIIKKA: PHP-tilastointiohjelmisto
Opinnäytetyö 55 s., joista liitteitä 26 s.
Huhtikuu 2012

Tämän opinnäytetyön tavoitteena oli suunnitella ja toteuttaa WEB-pohjainen helppokäyttöinen tilastointiympäristö salibandyjoukkueille.

Työssä käytettiin vain ilmaiseksi saatavilla olevia ohjelmia, jotta ohjelmaa käyttävä taho ei joutuisi maksamaan ohjelmien lisenssimaksuja.

Ohjelmisto suunniteltiin toimimaan Apache HTTP-serverillä, johon on asennettu MySQL- ja PHP-tuet. Tällöin sitä pystytään käyttämään lähes mistä vain, kunhan käyttäjällä on internet-yhteys ja internet-selain.

ABSTRACT

Tampere University of Applied Sciences
Degree Programme in Information Technology
Embedded Systems and Electronics

TIKKA MIIKKA: PHP statistics software

Bachelor's thesis 55 pages, appendices 26 pages
April 2012

The main goal to this thesis was to design and develop WEB-based easy to use statistics environment to floorball teams.

In this thesis was used only softwares which are free of charge, that everyone who wants to use the programs don't have to pay expensive license royalties.

The software was designed to run on Apache server program that has installed MySQL and PHP. Therefore it can be used almost everywhere as long as user has internet connection and web browser.

Keywords: PHP, MySQL, Apache, statistics

SISÄLLYS

1 JOHDANTO	6
2 OHJELMISTOT	7
2.1 Apache.....	7
2.2 PHP	8
2.3 MySQL.....	9
3 SUUNNITTELU	9
4 PHP-OHJELMA	10
4.1 Ottelut.....	11
4.2 Sarjataulukot	12
5 TIETOKANNAT	16
5.1 Ottelutietokanta	16
5.2 Pelaajatietokanta	16
5.3 Jäähy- tai maalitietokanta.....	17
5.4 Pistepörssit	17
5.6 Pelaajalistat	18
6 LÄHDEKOODI	18
6.1 Funktiot ja ohjelman alustus	19
6.2 Pelaajatilastot	24
6.3 Ottelutilastot.....	24
6.5 Pelaajakortti.....	26
7 TESTAUS	26
8 TIETOTURVA	27
9 YHTEENVETO	28
LÄHTEET.....	29
LIITTEET	30
Liite 1. PHP-pääohjelman lähdekoodi	30

LYHENTEET JA TERMIT

PHP	Hypertext Preprocessor, PHP on komentosarjakieli, jossa ohjelmakoodi tulkitaan vasta ohjelman suorituvaiheessa.
MySQL	Relaatiotietokantaohjelmisto
Apache	Apache HTTP Server on avoimeen lähdekoodiin perustuva HTTP-palvelinohjelma
WEB	Internet-verkko
.htaccess	Apache-palvelimen kansiokohtainen käyttäjälistaus
.htpasswd	Apache-palvelimen kansiokohtainen salasana-tiedosto

1 JOHDANTO

Työn tarkoituksena oli suunnitella ja toteuttaa salibandyjoukkueille helppokäyttöinen WEB-pohjainen tilastointiohjelmisto. Yli kymmenen vuoden salibandyn seuratoimintakokemuksella on huomattu, että sarjojen pyörittämisestä vastaavalla salibandyliitolla ei ole kunnollista, ajan tasalla olevaa tilastointijärjestelmää.

Monet joukkueet haluaisivat saada ajan tasalla olevaa tilastointia otteluistaan viimeistään ottelua seuraavana päivänä. Lähes jokaisella joukkueella on otteluiden tilastointia kotisivuillaan, mutta selkeää yhtenäistä kaikkia tietoja sisältävää tilastointia ei kuitenkaan löydy. Tästä syystä syntyi oman tilastointiohjelman tarve, jota joukkueiden aktiivihenkilöt voisivat käyttää. Tällöin saataisiin heti ottelutapahtumien jälkeen kaikki tilastot päivitettyä kaikkien halukkaiden nähtäville.

Työ painottuu ohjelmiston suunnitteluun ja toteutukseen. Työssä kerrotaan myös ohjelmiston toiminnasta sekä käytetyistä ohjelmista. Suunnittelussa tärkeänä suunnittelukriteerinä oli se, että ohjelmistoa pystyy käyttämään useampi käyttäjä samaan aikaan, jotta tilastot saataisiin nopeasti tallennettua tietokantaan.

Ohjelmistoalustaksi valittiin PHP-ohjelmointikieli sen tietokantayhteyksien helppokäyttöisyyden sekä sen dynaamisuuden vuoksi. Ohjelmisto toimii Apache HTTP-palvelinohjelmalla, johon on asennettu PHP-tuki. Ohjelma tallentaa kaikki tiedot MySQL-tietokantaan, josta niitä käyttäjän hakukriteerien mukaan haetaan.

2 OHJELMISTOT

Seuraavassa käydään läpi, mitä ohjelmia on käytetty työn tekemiseen ja mitä valmis ohjelma käyttää toimiakseen. Ohjelman tekemiseen käytettävien ohjelmien määrää rajoitti se, että työ tehtiin Windows-käyttöjärjestelmällä varustetulla tietokoneella. Itse ohjelma ei ole käyttöjärjestelmäriippuvainen, vaan vaatii toimiakseen internet-yhteydellä varustetun tietokoneen, jossa on internet-selain.

Ohjelmistovalinnoista ensimmäiseksi piti valita käytettävä ohjelmointikieli. Valituksi tuli PHP. Seuraavaksi valittavaksi tuli palvelinohjelmiston valinta, johon vaihtoehtoja ei ollut oikeastaan kuin kaksi: Apache Software Foundationin kehittämä Apache sekä Microsoftin kehittämä Microsoft Server. Microsoft server-ohjelmiston maksullisuuden vuoksi palvelinohjelmaksi tuli valituksi Apache. Tietokantaohjelman valinnaksi MySQL oli selkeä valinta, koska sen käytöstä oli omakohtaista kokemusta jo valmiiksi.

2.1 Apache

Apachen käyttö alkoi lataamalla ohjelman kotisivuilta viimeisin Windows-yhteensopiva versio Apache HTTP-palvelinohjelmasta. Ladatusta tiedostosta ohjelman asennus oli hyvin suoraviivaista. Asennusohjelmalle annettiin tiedot palvelimen nimestä ja verkko-osoitteesta sekä valittiin ohjelmalle haluttu ohjelmistopolku, johon ohjelma asennettiin.

Asennuksen jälkeen täytyi vielä muokata Apachen konfigurointitiedostoa httpd.conf. Tiedostosta luetaan ne asetukset, joilla Apache-palvelinohjelman halutaan joka käynnistyskerralla toimivan. Ensiasennuksen yhteydessä ei tehty isoja muutoksia, ainoastaan muutettiin päähakemiston polku.

PHP-ohjelmiston asennuksen jälkeen httpd.conf -tiedostoa piti muokata siten, että Apache tietää, mihin se ohjaa .php-päätteisten tiedostojen käsittelyn sekä mistä osoitteesta se lataa käynnistyessään PHP-tiedostojen käsittelyyn tarvittavaan .dll-moduulin.

Ohjelmiston kehittämisen aikana jouduttiin tekemään vielä yksi muutos `httpd.conf`-tiedostoon. Tällä muutoksella otettiin käyttöön Apache-palvelimen pää- ja alikansioiden luku- ja kirjoitusoikeuksien määrittäminen. Oikeuksia hallitaan `.htaccess`- ja `.htpasswd`-tiedostoilla. Muutos tehtiin, jotta ottelutietojen lisääminen ja muokkaaminen onnistuisi vain tilastointiohjelmaan salasanan saaneilta käyttäjiltä.

2.2 PHP

Ohjelma toteuttamiseen on valittu PHP-ohjelmointikieli, joka on suoritusvaiheessa tulkittava ohjelmointikieli. Ohjelman käytön aloitus tapahtui purkamalla `.zip`-päätteinen asennuspaketti PHP-ohjelmaa varten tehtyyn kansioon. Ohjelman asennuksessa ei tarvitse käyttää valmista Windows-asennuspakettia. Manuaalinen asennus riittää, koska Apache-palvelimen `httpd.conf`-tiedostoon pitää tehdä Apache-ohjelmistoa esiteltäessä kerrotut muutokset, joiden avulla PHP-tuki saadaan käyttöön.

PHP on avoimeen lähdekoodiin perustuva, joten sen käytöstä ei tarvitse maksaa lisenssimaksuja. PHP mahdollistaa ohjelmiston dynaamisuuden, jolloin selaimelle tulostettavat tiedot muuttuvat sitä mukaa, kun tilastointiohjelmaan tallennetaan lisää ottelutilastoja tai käyttäjä muuttaa haettavien tietojen hakuparametrejä.

PHP:n valintaan ohjelmointikieleksi vaikutti siihen sisäänrakennetut helppokäyttöiset tietokantayhteydet, joilla tietojen vieminen ja hakeminen tietokannasta sujuu helposti ja vaivattomasti. Tärkeänä seikkana PHP:n valinnalla käytettäväksi ohjelmointikieleksi oli myös sen käyttäjien aktiivinen käyttäjäyhteisö, josta mahdollisten ongelmatilanteiden syntyessä ohjelmointivirheen korjaamiseen, avun saaminen olisi tarvittaessa nopeaa. Mahdollinen avun tarve työn alussa piti ottaa huomioon, koska aikaisempaa kokemusta PHP-ohjelmoinnista ei ollut.

2.3 MySQL

Ohjelman tietojen tallentamiseen käytettiin MySQL-tietokantaa, joka sekin on avoimeen lähdekoodiin perustuva tietokantaohjelmisto. MySQL on relaatiotietokantaohjelmisto, jota kehittää MySQL AB.

MySQL valittiin tietokantaohjelmaksi soveltuvuutensa PHP- ja Apache-ohjelmistojen kanssa sekä siksi, että sen käytöstä oli ennakkoon jo hieman kokemusta, jolloin sen käytön opetteluun ja asentamiseen ei menisi liikaa aikaa. Apachen konfigurointi ja PHP-ohjelmoinnin opettelu veisi kuitenkin oman aikansa.

3 SUUNNITTELU

Työn tekeminen alkoi suunnitteleamalla, mitä tietoja salibandyottelun pöytäkirjasta pitää saada sisällytettyä mukaan ohjelmaan ja sitä kautta tallennettua tietokantaan.

Suunnittelussa päädyttiin siihen, että kaikki pöytäkirjassa oleva tieto tulitaisiin tallentamaan tietokantaan. Syynä tähän on se, että jos ottelupöytäkirjasta tarvittaisiin joskus jotain tilastoinnin kannalta epäoleellista tietoa, niin sekin olisi tallessa tai jos alkuperäisestä pöytäkirjasta tarvittaisi tehdä kopio niin se voitaisiin tehdä tallennettujen tietojen pohjalta menettämättä mitään ottelupöytäkirjaan kirjattua tietoa.

Seuraavaksi suunniteltiin, miten tietokantaa käytettäisiin. Tulisivatko kaikki tiedot samaan taulukkoon vai jaettaisiinko tiedot useampiin taulukoihin. Tiedot päädyttiin tallentamaan useaan eri taulukkoon. Taulukot jaettiin siten, että jokaiseen taulukkoon tulisi jokin tietty osa ottelupöytäkirjassa olevasta tilastoitavasta tiedosta. Ottelun perustiedot tallennetaan omaan taulukkoonsa, jolloin tallennettujen ottelujen tarkastelu onnistuisi lukemalla tiedot suoraan yhdestä taulukosta. Ottelun maali- ja jäähytiedot tallennettaisiin omaan taulukkoonsa. Kolmas taulukko, johon tietoja ottelusta tallennetaan, on pelaajatietotaulukko. Tähän taulukkoon tallennetaan pelaajan yksilöllinen tieto otteluista. Pelaajatietotaulukkoa käytetään laskettaessa ohjelmassa pistepörssejä.

3.1 PHP-ohjelman suunnittelu

Suunnittelun päälähtökohtana oli se, että ohjelmistoa käyttävän seuran ei tarvitse käyttää mitään muuta ohjelmistoa saadakseen internet-sivuilleen tilastot sarjataulukoista, otteluraporteista sekä kaikista muista mahdollisista pistepörsseistä ja pelatuista salibandyotteluista.

Tärkeänä lähtökohtana suunnittelussa pidettiin myös sitä, että tietoja voidaan hakea useilla eri hakuparametreilla, pelatun kauden, lohkon, joukkueen tai ottelun mukaan. Laajat hakuparametrivaihtoehdot antaisivat mahdollisuuden useiden joukkueiden käyttää ohjelmistoa riippumatta joukkueen sarjatasosta. Ohjelmistossa on myös hakutoiminnossa lisähakuparametri. Hakutuloksen ollessa vieläkin liian laaja ensimmäisen haun jälkeen voi lisähakuparametrillä tarkentaa haettuja tietoja, jotta haluttu tieto varmasti löytyisi.

4 PHP-OHJELMA

Seuraavassa osassa esitellään ohjelman ominaisuuksia sekä toimintaa kertomalla niistä ilman lähdekoodia (liite 1). Tässä osassa keskitytään ainoastaan ohjelman toiminnan sekä sen ominaisuuksien esittämiseen.

Ohjelma rakentuu kolmesta eri päävalinnasta: sarjataulukot, ottelut ja pelaajatilastot. Päävalinnalla vaikutetaan siihen, minkä tyyppisiä tietoja milläkin kerralla tulostetaan selaimen käyttäjälle nähtäväksi. Lisäksi ohjelmassa on admin-valikko, jossa lisätään ja muokataan ottelutilastoja sekä pelaajalistoja tietokantaan.

KUVA 1. Ohjelman alkuvalikko

Kuvassa 1 näkyvässä päänäkyvässä valitaan, mitä tilastoja halutaan tarkastella. Päävalintana olevasta tilastojenvalinta-alasvetovalikosta valitaan tilastotyyppi. Tilastoja katsottaessa sarjataso-alasvetovalikosta voi olla kerrallaan valittuna vain yksi sarjataso. Lohko-, kausi-, ottelu- ja joukkue-valinnoissa valittuina voi olla kaikki-valinta tai sitten CTRL-näppäin pohjassa voidaan tehdä useita valintoja samalla kerralla. Jos jossain monivalintavalikossa kaikki-valinta jää valituksi muiden valintojen kanssa, hakua tehtäessä valintana käytetään kaikki-valintaa eikä muista valinnoista välitetä. Kausi-, ottelu- ja joukkue-monivalintakentän alla oleviin ei valittu-alasvetovalikkoihin tulee valittavaksi ensimmäisen haun jälkeen haun parametrit täyttävät kaudet, ottelut ja joukkueet. Edellä mainitut valinnat toimivat jokaisella päävalinnalla.

Pelaajatilaston valintoina toimivat pörssit- ja lajittelut-alasvetovalikot. Pelaajat-alasvetovalikosta on valittuna aina kaikki-valinta, eikä pelaajat-valinnan lisävalinta ole käytössä. Päänäkymän alaraunassa on valintalaatikko valinnat ylivoima- ja alivoimapistepörssien sekä rangaistuslauksien tulostuksien näkymiseen pelaajatilatoissa. Oletusarvoisesti ne ovat poissa näkyvistä.

4.1 Ottelut

Ottelut-valinta valittuna ohjelma tulostaa näytölle hakukriteerien mukaiset otteluraportit. Jos otteluraportteja on enemmän kuin neljä kappaletta, ne lajitellaan siten, että niitä on aina neljä per sivu. Tämä siksi, että sivun selaaminen ylhäältä alas olisi sujuvaa eikä sivun lataus kestäisi pitkään. Jos otteluraportteja on enemmän kuin yksi sivu, laittamalla kursori seuraavaa sivua osoittavan linkin päälle ponnahtaa esiin infoteksti, josta näkee linkin takana olevat ottelut tuloksineen.

Otteluraporttiin tulostetaan ottelutiedoista oleellinen tieto nähtäväksi. Niitä ovat ottelussa pelaavat joukkueet, ottelun voittaja, ottelun paikka- ja aikatiedot sekä ottelussa tapahtuneet maalit ja jäähyt. Maalit ja jäähyt on lajiteltu erittäin siten, että joka erän maalitietojen perään tulee siinä erässä otetut jäähyt rangaistukset-osioon.

Ottelu: KISPO - TAMKO 0 - 8		Voittaja: TAMKO	
Päivämäärä: 11.12.2005	Pelipaikka: Tampere, Kaukajärvi	Alkamisaika: 15:19	Loppumisaika: 16:17
Kausi: 2005-2006 Runkosarja	KISPO aikalisä: 38:14	Tuomarit: Joonas Leponen ja Kari Karjalainen	TAMKO aikalisä: :-
1. era: 0 - 2			
Maalittiedot:			
04:43 TAMKO YV	0-1 Mylläri Tomi (Palosaari Juhho)		
12:37 TAMKO YV	0-2 Palosaari Juhho (Mylläri Tomi)		
Rangaistukset:			
04:42 KISPO	Mäkinen Joonas, 2 min. mailaan lyöminen tai potkaiseminen		
12:11 KISPO	Hirvonen Olli, 2 min. mailan painaminen tai sitominen		
2. era: 0 - 2			
Maalittiedot:			
15:16 TAMKO	0-3 Lahtinen Mika (Palosaari Juhho)		
23:03 TAMKO	0-4 Palosaari Juhho (Tapola Mikko)		
3. era: 0 - 4			
Maalittiedot:			
35:12 TAMKO	0-5 Miikka Tikka (Tapola Mikko)		
37:20 TAMKO	0-6 Viikman Marko (Lahtinen Mika)		
38:14 TAMKO	0-7 Tapola Mikko (Palosaari Juhho)		
39:33 TAMKO	0-8 Marttala Jonas (Viikman Marko)		

KUVA 2. Otteluraportti

Otteluraportissa on myös lisäominaisuuksia, joilla saa lisätietoja ottelusta ja siinä pelanneista pelaajista. Lisäominaisuudet on selitetty seuraavaksi.

Ottelussa pelanneiden joukkueiden nimenä olevaa linkkiä painaessa aukeaa uusi ikkuna, josta näkee tiivistelmän joukkueen pelaajista, pelinnumeroista, maalivahti- ja kapteenimerkinnästä sekä pisteistä siinä ottelussa.

Otteluraportissa esiintyvän pelaajan nimenä olevaa linkkiä painamalla aukeaa uusi sivu, johon tulostuu kyseisen pelaajan pelaajakorttisivu, josta näkee pelaajan tilastot kaikista tietokannasta löytyvistä otteluista, joissa pelaaja on pelannut.

4.2 Sarjataulukot

Sarjataulukko-valinta valittuna selaimen tulostuu hakukriteerien mukaisten otteluiden sarjataulukko, jossa joukkueet on lajiteltu ensin pisteiden, ja mikäli pisteet ovat yhtä suuret niin sen jälkeen, maalieron mukaisesti. Jos maalierokin on sama, niin silloin edelle menee joukkue, jolla on enemmän tehtyjä maaleja.

Nro:	Joukkue:	Ottelut:	Voitot:	Tasapelit:	Tappiot:	Tehd.Maal.:	Pääst.Maal.:	Maaliero:	Pisteet:
1	TAMKO	3	3	0	0	15	2	13	6
2	Punakone 2	1	0	0	1	1	3	-2	0
3	SHOWTIME	1	0	0	1	1	4	-3	0
4	KISPO	1	0	0	1	0	8	-8	0

KUVA 3. Sarjataulukko

4.3 Pelaajatilastot

Pelaajatilastot-valinta valittuna selaimelle tulostuu hakukriteerien mukaisissa otteluissa pöytäkirjaan merkityt pelaajat ottelukohtaisilla tilastoilla. Jos otteluita on valittuna useita eikä pistepörssi-valinta ole valittuna, pelaajatietoja tulostuu sivulle paljon.

Nro:	Nimi:	Joukkue:	Ottelut:	Maalit:	Syötöt:	Pisteet:	Pist./Ott.:	Jäähyt:	Min./Ott.:
1	Palosaari Juh	TAMKO	2	4	3	7	3.50	0	0.00
2	Tapola Mikko	TAMKO	3	2	2	4	1.33	0	0.00
3	Mylläri Tomi	TAMKO	3	1	3	4	1.33	0	0.00
4	Miikka Tikka	TAMKO	3	2	0	2	0.67	0	0.00
5	Lahtinen Mika	TAMKO	3	1	1	2	0.67	0	0.00
6	Vilkman Marko	TAMKO	3	1	1	2	0.67	0	0.00
7	Marttala Jonas	TAMKO	3	1	1	2	0.67	2	0.67
8	Kytölä Pasi	TAMKO	3	0	2	2	0.67	0	0.00
9	Virtanen Jukka	SHOWTIME	1	1	0	1	1.00	0	0.00
10	Johansson Petri	Punakone 2	1	1	0	1	1.00	2	2.00

KUVA 4. Pistepörssi

Ohjelmassa on omat valintavaihtoehdot, joita käytetään vain pelaajatilastoille. Näillä valinnoilla päätetään, miten pelaajatiedot valituista otteluista lajitellaan. Jos pörssi-valinta on valittuna, pelaajat tulostetaan valitun pörssivalinnan mukaisesti siten, että pelaajat tulostetaan laskevassa järjestyksessä siitä pelaajasta lähtien, jolla on eniten pisteitä valitussa kategoriassa. Jos valittuna kategoriana on pistepörssi, YV-pistepörssi tai AV-pistepörssi ja pelaajilla on yhtä paljon pisteitä, edelle menee se, kenellä on enemmän maaleja.

Yli- ja alivoima sekä rangaistuslaukaustilastot tulevat näkymään vain, jos niitä merkitsevät valintaruudut ovat valittuina ennen tilastohaun tekemistä. Tämä on tehty sen takia, että sivua on helpompi lukea ja näitä tilastoja käytetään harvemmin, joten ne eivät ole oletusarvoisesti valittuina.

Nro:	Nimi:	Joukkue:	Ottelut:	Maalit:	Syötöt:	Pisteet:	Pist./Ott.:	Jäähyt:	Min./Ott.:	YVM:	YVS:	YVP:
1	Palosaari Juh	TAMKO	2	4	3	7	3.50	0	0.00	2	1	3
2	Mylläri Tomi	TAMKO	3	1	3	4	1.33	0	0.00	1	2	3
3	Virtanen Jukka	SHOWTIME	1	1	0	1	1.00	0	0.00	1	0	1
4	Silvennoinen Vesa	TAMKO	3	1	0	1	0.33	0	0.00	1	0	1
5	Tapola Mikko	TAMKO	3	2	2	4	1.33	0	0.00	1	0	1
6	Jauhainen Juha	SHOWTIME	1	0	1	1	1.00	0	0.00	0	1	1
7	Markkanen Arttu	TAMKO	3	0	1	1	0.33	0	0.00	0	1	1
8	Kytölä Pasi	TAMKO	3	0	2	2	0.67	0	0.00	0	1	1

KUVA 5. Ylivoima-pistepörssi

4.4 Admin-valikko

Tilastointiohjelmaan uusien pöytäkirjojen täyttämistä ja vanhojen muokkaamista varten on suunniteltu erillinen Admin-valikko. Valikossa on myös joukkuekohtaisten nimilistojen täyttö- ja muokkausvalinnat. Valikko on salasanasuojauksen takana, joten ainoastaan pääkäyttäjällä sekä ottelupöytäkirjoja täyttävillä henkilöillä on pääsy sinne.

Admin-valikko		
Pyökirjavaihtoehdot:	Uuden pöytäkirjan täyttö	Vanhan pöytäkirjan muokkaus
Nimilistavaihtoehdot:	Uuden nimilistan täyttö	Vanhan nimilistan muokkaus

KUVA 6. Admin-valikko

Nimilistoja voidaan käyttää uuden pöytäkirjan täytön nopeuttamiseksi, jolloin pöytäkirjaa täyttävän henkilön ei tarvitse aina uutta pöytäkirjaa täyttäessään kirjoittaa pelaajien ja toimihenkilöiden nimiä, vaan ne tulevat valitun nimilistan mukaan automaattisesti uuteen pöytäkirjaan. Jokaisella joukkueelle voi tarvittaessa olla useampikin nimilista käytössään.

Nimilistan muokkaustoiminnolla voidaan nimelistasta korjata kirjoitusvirheitä sekä ottaa pois tai lisätä pelaajia tai toimihenkilöitä.

Ottelu lomakkeen esitiedot:			
Kotijoukkue:			
Valmiina olevat pelaajalistat:	TAMKO,2005-2006,Miesten 5, Divari ▾	Jos muu niin mikä:	
Tarvittava tyhjien pelaajarivien määrä:	5 ▾	Tarvittava tyhjien toimihenkilö rivien määrä:	0 ▾
Vierasjoukkue:			
Valmiina olevat pelaajalistat:	Ei ole listalla/muu ▾	Jos muu niin mikä:	
Tarvittava tyhjien pelaajarivien määrä:	20 ▾	Tarvittava tyhjien toimihenkilö rivien määrä:	3 ▾
<input type="button" value="OK"/>			

KUVA 7. Ottelun esitiedot

Pöytäkirjan täyttö alkaa valitsemalla nimilista koti- ja vierasjoukkueelle, jos sellaiset on tallennettu etukäteen. Vielä ennen pöytäkirjan täytön aloittamista täyttäjä valitsee, kuinka monta tyhjää riviä tarvitaan kummankin joukkueen pelaaja- sekä maalitiedoille. Tämä valinta tehdään, vaikka nimilista olisi valittunakin, koska täyttäjä voi haluta lisätä yksittäisiin otteluihin eri pelaajia listan ulkopuolelta, eikä listaa näin ollen tarvitse muuttaa joka kerta, vaan voidaan vain lisätä halutut henkilöt pöytäkirjaan tyhjille

riveille. Pöytäkirjan täyttö tapahtuu kirjoittamalla ottelupöytäkirjan paperisesta versiosta vastaavaan ohjelman ottelulomakekenttään. Ottelun yleisissä tiedoissa on joukkokenttiä, jotka pitää olla täytettyinä tai ohjelma ei anna tallentaa pöytäkirjaa tietokantaan. Nämä kentät ovat voittaja, joukkueiden nimet, lopputulos sekä ottelun päivämäärä. Kentät on valittu siten, että ottelut saadaan eroteltua toisistaan ohjelman tietokantahauissa.

Tilastosovellus, Ottelutietojen muokkaus	
Ottelut:	<input type="button" value="OK"/>
TAMKO - SHOWTIME, 14.12.2005, 2005-2006 Runkosarja ▼	

KUVA 8. Ottelunmuokkauksen-valinta

Ottelutietoja on mahdollista muokata oman valintansa kautta. Ensin valitaan listasta minkä ottelun tietoja halutaan muuttaa.

Tilastosovellus, Ottelutietojen muokkaus	
Ottelut:	
TAMKO - SHOWTIME, 14.12.2005, 2005-2006 Runkosarja ▼	
Kotijoukkueella: TAMKO = 15 Pelaajaa, jos haluat lisätä niitä valitse tästä:	- ▼
Kotijoukkueella: TAMKO = 0 toimihenkilöä, jos haluat lisätä niitä valitse tästä:	- ▼
Vierasjoukkueella: SHOWTIME = 15 Pelaajaa, jos haluat lisätä niitä valitse tästä:	- ▼
Vierasjoukkueella: SHOWTIME = 0 toimihenkilöä, jos haluat lisätä niitä valitse tästä:	- ▼
<input type="button" value="OK"/>	

KUVA 9. Ottelunmuokkauksen lisävalinta

Tämän jälkeen voidaan valita lisää tyhjiä rivejä, jos ottelupöytäkirjaan halutaan lisätä pelaajia tai toimihenkilöitä. Lopuksi aukeaa tietokannassa olevilla ottelutiedoilla täytetty ottelulomake, johon voidaan tehdä halutut muutokset ja tallentaa ottelupöytäkirja sen jälkeen takaisin tietokantaan.

5 TIETOKANNAT

Seuraavaksi esitellään, mitä tietoja tietokannan eri taulukot pitävät sisällään.

5.1 Ottelutietokanta

Ottelutietokantaan tallennetaan ottelutapahtuman yleiset tiedot siitä, missä ja milloin ottelu pelattiin ja ketkä siinä pelasivat. Ottelutietoja tallennettaessa jokainen ottelu saa oman yksilöllisen ottelun ID-numeron, jolla kaikki tiedot samasta ottelusta on jälkepäin helppo hakea eri tietokannoista sen perusteella.

5.2 Pelaajatietokanta

Tietokantaan tallennetaan jokainen ottelupöytäkirjaan merkitty henkilö, pelaajat sekä toimihenkilöt, joihin kuuluvat valmentajat, huoltajat ja mahdolliset lääkärit.

Juokseva	Nimi	Numo	Kaptr	Maalivahti	Maalit	Syötöt	Pisteet	Jaahyt	YVMaalit	YVSyötöt	YVPisteet	AVMaalit	AVSyötöt	AVPisteet	HLYritykset	HLMaalit	Ottelu	Toimihenkilö
▶	132	Saarkoski Jyri	1	-	-	0	0	0	0	0	0	0	0	0	0	0	Punakone 2 - TAMKO	0
	133	Syvänen Pasi	6	C	-	0	0	0	0	0	0	0	0	0	0	0	Punakone 2 - TAMKO	0
	134	Myllys Timo	7	-	-	0	0	0	0	0	0	0	0	0	0	0	Punakone 2 - TAMKO	0
	135	Stortz Stefan	8	-	-	0	0	0	0	0	0	0	0	0	0	0	Punakone 2 - TAMKO	0
	136	Backholm Mika	9	-	-	0	0	0	0	0	0	0	0	0	0	0	Punakone 2 - TAMKO	0
	137	Saarkoski Joni	10	-	-	0	0	0	2	0	0	0	0	0	0	0	Punakone 2 - TAMKO	0
	138	Still Hans	11	-	-	0	0	0	2	0	0	0	0	0	0	0	Punakone 2 - TAMKO	0
	139	Mäkelä Markus	13	-	-	0	0	0	0	0	0	0	0	0	0	0	Punakone 2 - TAMKO	0
	140	Kiviahho Henrik	16	-	-	0	0	0	0	0	0	0	0	0	0	0	Punakone 2 - TAMKO	0
	141	Mäki Kari	19	-	-	0	1	1	0	0	0	0	0	0	0	0	Punakone 2 - TAMKO	0
	142	Johansson Petri	20	-	-	1	0	1	2	0	0	0	0	0	0	0	Punakone 2 - TAMKO	0
	143	Haapakoski Timo	30	-	-	0	0	0	0	0	0	0	0	0	0	0	Punakone 2 - TAMKO	0
	144	Salin Oskari	41	-	-	0	0	0	0	0	0	0	0	0	0	0	Punakone 2 - TAMKO	0
	145	Mäkelä Kalle	42	-	MV	0	0	0	0	0	0	0	0	0	0	0	Punakone 2 - TAMKO	0
	146	Jääskeläinen Esa	72	-	-	0	0	0	0	0	0	0	0	0	0	0	Punakone 2 - TAMKO	0
	147	Niemenpää Tero	80	-	-	0	0	0	2	0	0	0	0	0	0	0	Punakone 2 - TAMKO	0
	148	Malinen Tori, TH															Punakone 2 - TAMKO	1
	149	Korpi Rauli, TH															Punakone 2 - TAMKO	1

KUVA 10. Ottelun pelaajatiedot-tietokanta

Pelaajan tiedoista löytyvät pelaajan maali- ja syöttöpisteet, pisteet yli- ja alivoimalla sekä tiedot rangaistuslaukauksista sekä jäähyminuuteista. Toimihenkilöillä muut kuin nimi- ja ottelu-kenttä jäävät tyhjiksi.

5.3 Jäähy- tai maalitietokanta

Jäähy- tai maalitietokantaan tallennetaan kaikki pelaajien tekemät maali- ja jäähytapahtumat otteluissa. Jokaisesta tapahtumasta riippumatta siitä, onko se jäähy tai maali, tietokantaan tallennetaan, millä sarjakaudella, milloin ja mitkä joukkueet siinä pelasivat. Myös tapahtuman aiheuttanut joukkue tallennetaan. Mikäli tapahtuma on maali, tallennetaan tieto maaliksi sekä maalitapahtumasta tulevat merkinnät, joita ovat maalin järjestysnumero, maalin syntyäika, maalintekijä, maalinsyöttäjä, tehtiinkö maali erikoistilanteesta, ottelun erä jossa maali syntyi sekä ottelutietokannasta tuleva ottelun ID-numero. Tapahtuman ollessa jäähy, tietokantaan tallennetaan tapahtuman merkinnäksi jäähy, jäähyyn saaneen pelaajan numero, jäähyyn pituus, jäähyyn aiheuttaneen rikkeen numeerinen syykoodi, jäähyyn alkamis- ja loppuaika, ottelun erä, jossa jäähy tapahtui sekä ottelutietokannasta tuleva ottelun ID-numero.

	Juokseva	Kausi	Paivamaara	Ottelu	Joukkue	JaahyVaiMaali	MJN	MAJMin	MTJSyy	MSJAn	MKJPa	MJera	PeliID
▶	39	2005-2006 Runkosarja	7.1.2006	Punakone 2 - TAMKO	TAMKO	M	1	12:56	19	11	YV	1	2
	40	2005-2006 Runkosarja	7.1.2006	Punakone 2 - TAMKO	TAMKO	M	2	14:41	9	12	YV	1	2
	41	2005-2006 Runkosarja	7.1.2006	Punakone 2 - TAMKO	Punakone 2	M	1	27:05	20	19	-	2	2
	42	2005-2006 Runkosarja	7.1.2006	Punakone 2 - TAMKO	TAMKO	M	3	29:16	9	20	-	2	2
	43	2005-2006 Runkosarja	7.1.2006	Punakone 2 - TAMKO	Punakone 2	J	80	2	67	12:33	12:56	1	2
	44	2005-2006 Runkosarja	7.1.2006	Punakone 2 - TAMKO	Punakone 2	J	10	2	93	13:18	14:41	1	2
	45	2005-2006 Runkosarja	7.1.2006	Punakone 2 - TAMKO	Punakone 2	J	11	2	71	22:14	24:14	2	2
	46	2005-2006 Runkosarja	7.1.2006	Punakone 2 - TAMKO	Punakone 2	J	20	2	71	44:20	45:00	3	2
	47	2005-2006 Runkosarja	7.1.2006	Punakone 2 - TAMKO	TAMKO	J	18	2	61	44:41	45:00	3	2
	48	2005-2006 Runkosarja	14.12.2005	TAMKO - SHOWTIME	TAMKO	M	1	23:36	25	20	YV	2	3
	49	2005-2006 Runkosarja	14.12.2005	TAMKO - SHOWTIME	SHOWTIME	M	1	30:40	13	8	YV	3	3
	50	2005-2006 Runkosarja	14.12.2005	TAMKO - SHOWTIME	TAMKO	M	2	35:14	17	18	-	3	3
	51	2005-2006 Runkosarja	14.12.2005	TAMKO - SHOWTIME	TAMKO	M	3	37:39	23	12	-	3	3
	52	2005-2006 Runkosarja	14.12.2005	TAMKO - SHOWTIME	TAMKO	M	4	39:22	21	-	-	3	3

KUVA 11. Jäähy- tai maalitietokanta

5.4 Pistepörssit

Pistepörssiä laskettaessa vertailtavien pelaajien määrä vaihtelee aina pistepörssityypistä ja ottelumäärästä riippuen, jonka vuoksi pistepörssien laskemista varten on käytössä tyhjä tietokanta. Sitä käytetään siten, että ohjelma lähettää siihen tallennettavat tiedot halutuissa otteluissa pelanneista pelaajista, joilla on pistemerkitöjä. Tallennuksen jälkeen pelaajat lajitellaan laskevasti halutun pistepörssiä varten. Lajittelun jälkeen pistepörssitiedot palautetaan ohjelmalle ja tietokanta tyhjenetään uutta hakua varten.

5.5 Jäähykoodit

Jäähykoodi-tietokantaan on tallennettu sanalliset selitykset ottelupöytäkirjassa oleville numeerisille jäähykoodeille. Tätä tietokantaa käytetään vain ottelupöytäkirjoja tulostettaessa.

	Juokseva	Koodi	Selitys
▶	1	52	epäurheilijamainen käytös
	2	71	estäminen
	3	93	huitominen
	4	54	häiritsevä valmentaminen
	5	50	kieltäytyminen lähtemästä jäähypenkiltä
	6	72	kiinnittäminen
	7	82	korkea jalka
	8	81	korkea maila

KUVA 12. Jäähykoodit-tietokanta

5.6 Pelaajalistat

Pelaajalista-tietokantaan voidaan tallentaa joukkuekohtaisia pelaajalistoja. Niiden käyttö helpottaa ja nopeuttaa ottelupöytäkirjojen täyttämistä, koska silloin ei tarvitse joka kerta kirjoittaa kaikkien pelaajien nimiä ja numeroita pöytäkirjaan, vaan ne voidaan ladata valmiiksi ennakkoon tallennetuista joukkueiden pelaajalistoista.

6 LÄHDEKODI

Seuraavaksi esitellään lähdekoodia. Lähdekoodi on yhdistelmä PHP-koodia ja HTML-koodia, koska ohjelma on suunniteltu toimimaan internet-selaimessa. Ohjelmassa on käytetty myös hieman javascriptiä.

Lähdekoodi jakautuu neljään osaan. Ensimmäinen osa sisältää HTML-sivun perustiedot ja kaikki ohjelmassa käytettävät funktiot. Ensimmäisessä osassa haetaan tietokannasta kaikkien ohjelmaan tallennettujen otteluiden tiedot ohjelman päänäkökulmaan. Toisessa osassa käsitellään kaikki pelaajatilastot-valinta valittuna olevat tapahtumat.

Kolmannessa osassa käsitellään ottelutilastot-valinnan tapahtumat. Viimeisessä, eli neljännessä osassa käsitellään sarjataukko-valinnan tapahtumat.

Lähdekoodissa oleva `<?php`-elementin alku ilmoittaa seuraavan osan olevan PHP-koodia. Koodi lähetetään PHP-tulkille, jonka käsittelyn jälkeen se lähetetään selaimelle. Näin toimitaan, kunnes koodissa tulee `?>`-lopetuselementti.

6.1 Funktiot ja ohjelman alustus

Ohjelman ollessa osaksi HTML-koodia sen alussa on HTML-sivun määrittelyt. HTML-koodissa elementit määritellään lähes poikkeuksetta peräkkäisillä kulmasulkein merkityillä tunnisteilla, joista jälkimmäinen alkaa vinoviivalla lopputunnisteen merkiksi. Selain ymmärtää tunnisteessa olevan tiedon mukaan, mitä tunnisteiden välisellä tiedolla tehdään. Esimerkiksi `<TITLE>Otsikko</TITLE>`-elementin väliin kirjoitettu teksti näkyy selaimen yläreunassa sivun otsikkona.

```
<html>
<head>
<meta http-equiv="Content-Type" CONTENT="text/html; charset=iso-8859-1">
<title>SB TILASTOT &#169; by Woody</title>
<link rel="icon" href="favicon.ico" type="image/x-icon">
<link rel="shortcut icon" href="favicon.ico" type="image/x-icon">
<link rel="stylesheet" type="text/css" href="style.css">
<SCRIPT Language="JavaScript" type="text/javascript">
<!--
var pelaajat;
var kp=new Array();
var vp=new Array();
function load(pelaajat)
{
    var NEW_WIN=null;
    NEW_WIN =
window.open(pelaajat, '', 'scrollbars=yes,menubar=no,height=460,width=300,resizable=
yes,toolbar=no,location=no,status=no;
}
// -->
// PRELOADING IMAGES
if (document.images)
{
    img_on =new Image(); img_on.src ="Reset3.gif";
    img_off=new Image(); img_off.src="Reset1.gif";
}
function handleOver()
{
    if (document.images) document.imgName.src=img_on.src;
}
function handleOut()
{
    if (document.images) document.imgName.src=img_off.src;
}
</Script>
</head>
```

Ohjelmassa käytetystä javascript-koodista näkyy tarvittavien muuttujien sekä funktion määrittely HTML-sivun otsikko-osassa. Lisäksi tilasto-ohjelman RESET-painikkeena olevat kuvat ladataan selaimen muistiin tässä vaiheessa.

Ohjelman rakenteen selkeyden vuoksi ohjelmatiedoston alussa on kaikki funktiot, joita ohjelmassa tarvitaan sen toimimiseen.

```
function SQLSort($vertailu,$Hakutieto,$haettava)
```

Ensimmäisenä on SQLSort-niminen funktio, jolla järjestellään MySQL:n avulla pistepörssit halutulla tavalla. Funktio saa kutsuttaessaan kolme parametria, joista ensimmäinen \$vertailu sisältää kaikki pelaajatiedot. Seuraava \$Hakutieto-parametri sisältää, minkä tiedon mukaan pelaajat lajitellaan. Viimeinen \$haettava-parametri on valinnainen. Sitä käytetään vain osassa hauista, joissa on mahdollista pelaajien olla samoissa pisteissä väärässä järjestyksessä. Funktio tarkistaa ensin tietokantayhteyden, jonka jälkeen se tallentaa kaikki funktioon tuotavat pelaajat yksi kerrallaan tietokantaan. Tämän jälkeen tiedot haetaan annettujen hakuparametrien mukaan tietokannasta ja tallennetaan taulukkomuuttujaan, jonka tieto palautetaan funktiota kutsuneeseen muuttujaan.

```
function Taulukointi($vertailu,$Hakutieto,$haettava)
{
    for($x=0;$x<count($vertailu);$x++)
    {
        if ($vertailu[$x]== trim($Hakutieto[$haettava]))
        {
            $temppe=1;
        }
    }
    if ($temppe==0)
    {
        $vertailu[] = trim($Hakutieto[$haettava]);
    }
    $temppe=0;
}
}
```

Taulukointi-funktio saa kutsuttaessaan kolme muuttujaparametria. Funktio ei palauta mitään, koska sitä kutsutaan viittausmuuttujalla. Funktiota käytetään tietokannasta haetun tiedon taulukoimiseen. Ensimmäinen muuttuja on funktiota kutsuvan taulukkomuuttujan viittaus. Toinen muuttuja on yksi tietokannasta saatu rivi, josta

haetaan kolmannen muuttujan mukaista tietokentän arvoa. Ensin lasketaan montako riviä taulukko sisältää, tämän jälkeen verrataan jokaista viitatus taulukon riviä tietokannasta tulevan rivin muuttujan arvoon. Jos uutta haettavaa tietoa ei löydy, funktio ei käytännössä tee mitään, eikä vertailtavaan taulukkomuuttujaan lisätä uutta tietoa. Uuden tiedon löytyessä se lisätään viitatus taulukon loppuun.

```
function Lista($lista,$valittu)
{
    for( $x=0;$x<count($lista);$x++)
    {
        if( $valittu == $lista[$x]) echo "<option
selected>$lista[$x]</option>";
        else echo "<option> $lista[$x] </option>";
    }
}
```

Lista-funktio on tehty vähentämään koodirivien lukumäärää. Funktio tekee tilasto-ohjelman pääsivun yksirivisten alasvetovalikkojen HTML-koodin. Funktio saa parametreinä kaksi muuttujaa. Lista-tilausta, jossa on jokaisen alasvetovalikon rivin tiedot sekä valittu-tieto, jossa on se taulukon rivi, joka on valittuna.

```
function isoLista($isoLista,$valittu)
{
    for( $x=0;$x<count($isoLista);$x++)
    {
        if ($valittu[0] == "") $valittu[0] = "Kaikki";
        for( $y=0;$y<count($valittu);$y++)
        if( $valittu[$y] == $isoLista[$x]) $Samat =1;
        if( $Samat == 1) echo "<option selected>$isoLista[$x]</option>";
        else echo "<option> $isoLista[$x] </option>";
        $Samat =0;
    }
}
```

Funktion-isolista tehtävänä on vähentää monivalintavalintojen tekemiseen tarvittavien koodirivien määrää. Funktio tekee saman, minkä lista-funktiokin, mutta monivalintavalintojen osalta.

```
function Jarjestely($Taulukko,$tyyli,$lisatyyli,$stosiliasatyyli)
```

Järjestely-funktiossa järjestetään tilastointiohjelman sarjataulukko-valinnan sarjataulukko halutunlaiseen muotoon. Funktio saa parametreinä neljä muuttujaa.

Ensimmäinen sisältää viittauksen kaikkiin sarjataulukon tietoihin. Loput muuttujat sisältävät tiedot, joiden mukaan joukkueet järjestellään. Joukkueet järjestellään siten, että joukkue, jolla on eniten pisteitä on ensimmäisenä. Pisteiden ollessa tasan joukkue, jolla on parempi maaliero on edellä. Maalieronkin ollessa sama edelle menee joukkue, joka on tehnyt enemmän maaleja. Mikäli joukkueet ovat vieläkin tasoissa, se kumpi joukkue on luettu ensin tietokannasta on edellä.

```
function Viiminen($haluttu,$sana)
{
    for($x=count($haluttu);$x>0;$x--)
        $haluttu[$x]=$haluttu[($x-1)];
    $haluttu[0] = $sana;
}
```

Funktiolla viiminen siirretään monivalintavalintataulukoiden ensimmäiseksi valinnaksi haluttu merkkijono.

Funktioiden esittelyn jälkeen lähdekoodissa seuraavaksi haetaan tietokannasta kaikki ottelutiedot ja taulukoidaan ne käyttämällä taulukointifunktiota. Taulukoinnin jälkeen tarkastetaan, ladataanko sivua ensimmäistä kertaa. Tämä tehdään katsomalla PHP:ssä valmiiksi olevalla isset-funktiolla, onko tilastojen valintamuuttujalle asetettu arvoa. Samalla tarkastetaan, onko kaikissa monivalintavalinnoissa valittuna kaikki-valinta. Jokaisen monivalinnan ollessa kaikki-arvossa, sekä jos isset-funktio palauttaa epäpositiivisen arvon, jatketaan lajittelemalla kaikki taulukoidut tiedot aakkosjärjestykseen niiden selaamisen helpottamiseksi.

Lisävalintojen tietokantahaut tehdään ainoastaan, jos jostakin monivalinnasta on valittuna jokin muu kuin kaikki-valinta, eikä kyseessä ole ohjelman ensimmäinen latauskerta. Haetut tiedot taulukoidaan taulukointi-funktiolla.

```
<form action="<?php $PHP_SELF ?>" method="<?php GET ?>">
```

Kaikkien tietojen tietokannasta hakemisen jälkeen tulostetaan tilastointiohjelman päävalikon HTML-koodi selaimelle. HTML-lomakkeen aktivoimisen jälkeiseksi osoitteeksi on valittu tilastointiohjelman oma osoite, jolloin se toimii aina samalta lomakkeelta. Lomakkeen lähetysmetodiksi on valittu GET-tyyppiset muuttujat eli muuttujien arvot luetaan selaimen osoiteriviltä.

```

if (count($Lohko_valinta)>0)
{
    if (count($Lohko_valinta)==1 && $Lohko_valinta[0] == "Kaikki") $Lohkot =
    "Kaikki";
    else
    {
        if( $Lohko_valinta[0] == "Kaikki")
        {
            $kysely .=" AND ( ";
            for($x=1;$x<count($Lohko_valinta);$x++)
            {
                $kysely .=" Lohko = '$Lohko_valinta[$x]' ";
                if(count($Lohko_valinta) >2 && $x<(count($Lohko_valinta)-
                1)) $kysely .=" OR ";
            }
        }
        else
        {
            $kysely .=" AND ( ";
            for($x=0;$x<count($Lohko_valinta);$x++)
            {
                $kysely .=" Lohko = '$Lohko_valinta[$x]' ";
                if(count($Lohko_valinta) >=2 && $x<(count($Lohko_valinta)-
                1)) $kysely .=" OR ";
            }
        }
        $kysely .=")";
    }
}

```

Tietokantakyselyn tekeminen monivalinnasta tapahtuu neljällä sisäkkäisellä ehtolausekkeella, jokaisella monivalinnalla on omansa. Tässä lohkonmonivalinta esimerkiksi otettuna. Ensin tarkastellaan, onko valintoja valittuna, tämä siksi, että monivalinnan voi jättää myös tyhjäksi. Seuraavaksi tarkastellaan valintoja, joissa on vain yksi valinta valittuna sekä tarkistetaan, onko tämä valinta kaikki. Seuravassa ehtolausekkeessa tarkastetaan, onko monivalinnan ensimmäinen valinta kaikki, jos on, niin se jätetään haussa huomioimatta, koska valinta jää käyttäjältä yleensä vahingossa päälle ja tällöin hakutulos ei ole käyttäjän haluama. Valitut lohkovalinnat lisätään \$kysely-nimiseen tekstityyppiseen muuttujaan.

6.2 Pelaajatilastot

Tilastovalinnasta pelaajatilastojen ollessa valittuna rakennetaan ehtolausekkeilla alaveto- ja monivalintavalinnoiden mukainen tietokantahaku. Pelaajatietojen osalta joudutaan käyttämään omaa vain tätä tilastotyyppiä varten tehtyä ehtolausekerakennetta. Syynä tähän on se, että pelaaja voi pelata ottelun kummassa joukkueessa tahansa, joten tietokantahaun hakulausekkeiden ehdot ovat erilaiset kuin muissa tilastotyypeissä. Pelaajatietokannasta palautuvasta hakutuloksesta tarkastetaan, että siinä löytyy vähintäänkin yhden ottelun tilastot. Mikäli ottelutietoja ei löydy tulostetaan virheilmoitus.

Seuraavaksi katsotaan, onko jokin pistepörssi valittuna. Jonkun pistepörssin oltuna valittuna lasketaan näytettäviin tilastoiden vain ne pelaajat, joilla on tilastomerkinlöjä valitussa pistepörssityypissä. Valitusta pistepörssistä merkinlöjä omaaville pelaajille lasketaan kaikki merkinnät, jonka jälkeen taulukko, jossa tiedot ovat, käsitellään SQLSort-funktiossa, josta ne palautuvat halutussa järjestyksessä. Saatuu valmis taulukko tulostetaan näytölle.

Pelaajatilastoihin on lisätty vielä mahdollisuus lajitella kaikki haettavat pelaajat minkä tahansa pelaajan tilastoissa olevan tiedon mukaan. Lajittelu tehdään automaattisesti suurimmasta pienimpään. Lajittelu toimii ainoastaan silloin, kun mitään pistepörssiä ei ole valittu.

6.3 Ottelutilastot

Ottelutilastojen tulostus aloitetaan tekemällä tietokantahaku ottelut-tietokannasta, päävalinnan valintojen mukaisesti. Tietokantahaun tulokset tallennetaan rivi kerrallaan yhteen taulukkomuuttujaan. Taulukosta tarkastetaan montako ottelua siihen hakuehdoilla tallennettiin. Ottelutietojen puuttuessa taulukosta tulostetaan virheilmoitus. Hakuehdoilla ottelutietoja voi löytyä yhdestä kaikkiin samalla sarjatasolla pelattuihin tietokannassa oleviin otteluihin. Tästä johtuen, jos otteluita on taulukossa yli neljä kappaletta, tulostuu päävalikon alle linkkilista. Jokaisessa linkissä on kommenttina lista sen takana olevista otteluista tuloksineen.



KUVA 13. Ottelulinkkilista

Ennen kuin ottelutiedot voidaan tulostaa otteluraportiksi pitää ottelutietoihin lisätä jäähy- vai maali-tietokannasta jokaiseen maali- ja jäähytilanteeseen tapahtuman tekijän nimi tai nimet. Tämän lisäksi jäähykoodit-tietokannasta haetaan jokaiseen ottelussa tulleetseen jäähyyn, sen numeeriselle syykoodille sanallinen selitys. Ottelutietojen tulostusta varten täytyy vielä laskea jokaisen ottelun jokaisessa erässä olevien maali- ja rangaistustapahtumien lukumäärät, jotta ne voidaan tulostaa omissa silmukkalausekkeissaan. Otteluraportin tulostus tapahtuu yhdessä silmukkalausekkeessa, joka toistetaan ottelussa olevien erien verran. Jos ottelu on mennyt jatkoajalle tai rangaistuslaukauskilpailuun asti, ne tulostetaan omissa silmukoissaan alkuperäisten erien perään.

6.4 Sarjataulukot

Sarjataulukot valittuna haetaan tietokannasta päävalinnan valintojen mukaisesti ottelutietokannasta hakutietoja vastaavat ottelut. Hakutulosten perusteella ensin tarkastetaan, onko hakutulosten perusteella yhtään ottelua valittuna. Jos otteluita ei ole valittuna, tulostetaan selaimelle virheilmoitus. Otteluiden löydyttyä laitetaan otteluissa pelanneet joukkueet taulukkoon. Taulukkoa tehdessä tarkastetaan, että jokainen joukkue voi olla vain kerran mainittuna. Joukkueiden taulukoimisen jälkeen lasketaan jokaiselle joukkueelle pisteet, tehdyt ja päästetyt maalit ottelutiedoista. Tiedot lasketaan siten, että voitosta saa kaksi pistettä, tasapelistä yhden ja tappiosta nolla pistettä. Laskemisen jälkeen kutsutaan funktiot-kappaleessa esiteltyä järjestely-funktiota, jossa joukkueet järjestellään oikeaan järjestykseen.

6.5 Pelaajakortti

Lähdekoodin lopussa olevan pelaajakortin tulostusosassa tulostetaan yksittäisen pelaajan kaikkien otteluiden tilastot kausittain jaoteltuina ja lopuksi kaikkien kausien tilastot yhteenlaskettuina. Pelaajakorttiin pääsee painamalla pelaajan nimenä olevaa linkkiä otteluraportissa tai pelaajatilastojen listauksessa.

Pelaajakorttia varten haetaan kaikista tietokannassa olevista peleistä ne, joissa valittu pelaaja on pelannut kyseisessä joukkueessa. Hakutiedosta etsitään ensin taulukointifunktiolla kaikki ne eri pelikaudet, joilla pelaaja on pelannut. Tämän jälkeen lisätään vain pelaajatietokannasta tehdyn haun perusteella kaikki tilastomerkinnot taulukkoon. Valmis taulukko tulostetaan näytölle. Tulostettaessa taulukkoa siihen tehdään kaksi käyttäjälle näkyvää muutosta. Pisteitä ja jäähyjä ottelua kohden olevat luvut tulostetaan kahdella desimaalilla.

7 TESTAUS

Ohjelma on testattu pääasiassa kokeilemalla kaikkia ominaisuuksia, jolloin mahdolliset virhetilanteet on huomattu.

Testauksen aikana löytyneistä virheistä ei ole laadittu erillistä testauspöytäkirjaa, vaan ne on korjattu välittömästi niiden löydyttyä.

Suurimmat ongelmat virheiden kanssa olivat eri merkistökoodausstandardien kanssa, koska tietokannan päivityksen yhteydessä huomattiin, että uuteen versioon ei pystytty kirjoittamaan kuin UTF8-muodossa olevaa dataa, jolloin kaikkiin tietokantaa koskeviin tietoihin piti tehdä tyyppimuunnokset ennen tietokantaan kirjoitusta sekä lukemisen jälkeen.

8 TIETOTURVA

Pöytäkirjojen täyttöoikeuden omaavilla on jokaisella oma käyttäjätunnuksensa ja salasanansa, jotta mahdollisissa väärinkäytöstilanteissa syyllisen selvittäminen olisi helppoa.

Tietoturvan kannalta suurin selkeä muutos oli Apache-palvelinohjelmassa oleva kansiokohtainen käyttöoikeuksien hallinnan käyttöönotto. Tällä toimenpiteellä tietokantaan kirjoitus saatiin salasanalla suojatuksi. Oikeuksia hallitaan .htaccess- ja .htpasswd-tiedostoilla, joiden näkyminen selaimelle on estetty Apachen asetuksissa. Näin saadaan muista ohjelmista riippumaton salanasuojattu kansioihin perustuva ympäristö. Tällä tavalla toteutettu salanasuojaus vaatii kuitenkin pääkäyttäjälle kaikki oikeudet Apache-palvelimen päähakemistoihin, joten pääkäyttäjän pitää olla todella huolellinen omasta tietoturvastaan.

9 YHTEENVETO

Työn tavoitteena oli suunnitella ja ohjelmoida tilastointiohjelmisto salibandyjoukkueille. Työn aiheen valintahetkellä salibandyliitolla ei ollut ajantasalla olevaa tilastointijärjestelmää alasarjajoukkueille, joten työn tavoitteena oli saada ainakin omalle joukkueelle kaikenkattava tilastointijärjestelmä. Ohjelman helppokäyttöisyyden vuoksi muutkin joukkueet pystyisivät halutessaan ottamaan sen käyttöönsä.

Työn alkuperäisen suunnitelman mukaan ohjelmoinnin suunniteltiin kestävän noin neljän viikon verran. Ohjelmointityö kesti kuitenkin lähes tuplasti tämän. Ohjelmointityöhön tarvittun ajan kasvu johtui osittain huonosta määrittelystä ja osittain kokemattomuudesta ohjelmistoprojekteista. Suunnitteluun ja määrittelyyn voitaisiin käyttää vielä enemmän aikaa.

Tietokantaohjelman päivitys kesken työn tekemisen tuotti myös vaikeuksia, koska päivityksen jälkeen tietokantaan ei pystynyt kirjoittamaan UTF8-merkistökoodauksella olevaa dataa. Tämän takia ohjelmaan piti tehdä useita muutoksia.

Työn graafiselle ulkoasulle voisi tehdä parannuksia. Työssä keskityttiin ohjelman toiminnallisuuteen eikä graafiseen ulkoasuun. Lopullinen ohjelma on toimiva ratkaisu jokaisen salibandyjoukkueen tilastojen ylläpitämiseen.

LÄHTEET

PHP-kotisivu. Päivitetty 08.04.2012. Luettu 08.04.2012.
<http://www.php.net/>.

MySQL-kotisivu. Päivitetty 01.04.2012. Luettu 08.04.2012.
<http://www.mysql.com/>.

Apache-kotivu. Päivitetty 01.04.2012. Luettu 08.04.2012.
<http://www.apache.org/>.

LIITTEET

Liite 1. PHP-pääohjelman lähdekoodi

Liite 1 1(26)

```

<html>

<head>
<meta http-equiv="Content-Type" CONTENT="text/html; charset=iso-8859-1">
<title>SB TILASTOT &#169; by Woody</title>
<link rel="icon" href="favicon.ico" type="image/x-icon">
<link rel="shortcut icon" href="favicon.ico" type="image/x-icon">
<link rel="stylesheet" type="text/css" href="style.css">
<SCRIPT Language="JavaScript" type="text/javascript">
<!--
var pelaajat;
var kp=new Array();
var vp=new Array();
function load(pelaajat)
{
    var NEW_WIN=null;
    NEW_WIN =
window.open(pelaajat,','scrollbars=yes,menubar=no,height=460,width=300,resizable=yes,toolbar=no,location=no,status=no');

}

// -->
// PRELOADING IMAGES
if (document.images)
{
    img_on =new Image(); img_on.src ="Reset3.gif";
    img_off=new Image(); img_off.src="Reset1.gif";
}

function handleOver()
{
    if (document.images) document.imgName.src=img_on.src;
}

function handleOut()
{
    if (document.images) document.imgName.src=img_off.src;
}
</Script>
</head>

<body class="taus">

<?php

//<script type="text/javascript"> <meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
//document.write("Viimeksi päivitetty: " + document.lastModified)
//</script>
//perustiedot tiedot tiedostosta
include'admin/tiedot.php';

include'admin/tietokanta.php';

//tarvittavat functiot
/*
function Sarjis($Taulukko,$Styyli,$lisatyyli,$stosilisatyyli)
{
    for($t=0;$t<count($Taulukko);$t++)
    for($x=1;$x<count($Taulukko);$x++)
    {
        if ($Taulukko[$x-1][$Styyli] < $Taulukko[$x][$Styyli] )
        {
            $temp =$Taulukko[$x];
            $Taulukko[$x]=$Taulukko[$x-1];
            $Taulukko[$x-1]=$temp;
        }
        if ($Taulukko[$x-1][$Styyli] == $Taulukko[$x][$Styyli] && $Taulukko[$x-1][$lisatyyli] <
        $Taulukko[$x][$lisatyyli])

```

Liite 1 2(26)

```

    {
        $temp = $Taulukko[$x];
        $Taulukko[$x] = $Taulukko[$x-1];
        $Taulukko[$x-1] = $temp;
    }
    if ($Taulukko[$x-1][Styyli] == $Taulukko[$x][Styyli] && $Taulukko[$x-1][Lisatyylit] ==
    $Taulukko[$x][Lisatyylit] && $Taulukko[$x-1][Stosilisatyylit] < $Taulukko[$x][Stosilisatyylit])
    {
        $temp = $Taulukko[$x];
        $Taulukko[$x] = $Taulukko[$x-1];
        $Taulukko[$x-1] = $temp;
    }
}
*/

// Alla olevalla funktiolla käytetään MYSQL:n sorttaus algoritmia ja sortataan pelaajalista halutun tiedon mukaan
function SQLSort($vertailu,$Hakutieto,$haettava)
{
    include 'admin/tietokanta.php';
    $sote = mysql_connect( $host , $kayttaja , $salasana );
    if ( ! $sote ) die( "Tietokantaan ei saatu yhteyttä!" );
    mysql_select_db( $tietokanta, $sote ) or die ( "Tietokantaa $tietokanta: "." ei saatu valittua".mysql_error() );
    for($x=0;$x<count($vertailu);$x++)
    {
        $Ni = $vertailu[$x][Nimi];
        $Ma = $vertailu[$x][Maalit];
        $Sy = $vertailu[$x][Syotot];
        $Pi = $vertailu[$x][Pisteet];
        $Ja = $vertailu[$x][Jaahyt];
        $Ot = $vertailu[$x][Ottelut];
        $YVMa = $vertailu[$x][YVMaalit];
        $YVSY = $vertailu[$x][YVSYotot];
        $YVPI = $vertailu[$x][YVPisteet];
        $AVMa = $vertailu[$x][AVMaalit];
        $AVSY = $vertailu[$x][AVSYotot];
        $AVPI = $vertailu[$x][AVPisteet];
        $RLYr = $vertailu[$x][RLYritykset];
        $RLMa = $vertailu[$x][RLMaalit];
        $Jo = $vertailu[$x][Joukkue];
        $kysely = "INSERT INTO pinnatemp (Nimi, Maalit, Syotot, Pisteet, Jaahyt, Ottelut, YVMaalit, YVSYotot, YVPisteet, AVMaalit, AVSYotot, AVPisteet, RLYritykset, RLMaalit, Joukkue, ID) values
        ('$Ni','$Ma','$Sy','$Pi','$Ja','$Ot','$YVMa','$YVSY','$YVPI','$AVMa','$AVSY','$AVPI','$RLYr','$RLMa','$Jo','1');";

        mysql_query( $kysely, $sote ) or die( "Pinnatietojen kirjoitus epäonnistui".mysql_error() );
    }

    if ($haettava == "") $kysely = "SELECT Nimi, Maalit, Syotot, Pisteet, Jaahyt, Ottelut, YVMaalit, YVSYotot, YVPisteet, AVMaalit, AVSYotot, AVPisteet, RLYritykset, RLMaalit, Joukkue FROM pinnatemp ORDER BY $Hakutieto DESC;";
    else $kysely = "SELECT Nimi, Maalit, Syotot, Pisteet, Jaahyt, Ottelut, YVMaalit, YVSYotot, YVPisteet, AVMaalit, AVSYotot, AVPisteet, RLYritykset, RLMaalit, Joukkue FROM pinnatemp ORDER BY $Hakutieto DESC, $haettava DESC, ($Hakutieto/ Ottelut) DESC;";
    $sqlhakutieto = mysql_query( $kysely, $sote );
    $x=0;
    while ($sqlrivi = mysql_fetch_array( $sqlhakutieto ))
    {
        $pelaajat[$x][Nimi] = utf8_decode($sqlrivi[Nimi]);
        $pelaajat[$x][Maalit] = $sqlrivi[Maalit];
        $pelaajat[$x][Syotot] = $sqlrivi[Syotot];
        $pelaajat[$x][Pisteet] = $sqlrivi[Pisteet];
        $pelaajat[$x][Jaahyt] = $sqlrivi[Jaahyt];
        $pelaajat[$x][Ottelut] = $sqlrivi[Ottelut];
        $pelaajat[$x][YVMaalit] = $sqlrivi[YVMaalit];
        $pelaajat[$x][YVSYotot] = $sqlrivi[YVSYotot];
        $pelaajat[$x][YVPisteet] = $sqlrivi[YVPisteet];
        $pelaajat[$x][AVMaalit] = $sqlrivi[AVMaalit];
        $pelaajat[$x][AVSYotot] = $sqlrivi[AVSYotot];
        $pelaajat[$x][AVPisteet] = $sqlrivi[AVPisteet];
        $pelaajat[$x][RLYritykset] = $sqlrivi[RLYritykset];
        $pelaajat[$x][RLMaalit] = $sqlrivi[RLMaalit];
        $pelaajat[$x][Joukkue] = utf8_decode($sqlrivi[Joukkue]);
        $x++;
    }
    $tiedot = "DELETE FROM pinnatemp WHERE ID='1'";
    mysql_query( $tiedot, $sote ) or die( "Ottelu tietojen poisto tietokannasta epäonnistui".mysql_error() );
    return $pelaajat;
}

```

Liite 1 3(26)

```

}

function Taulukointi($vertailu,$Hakutieto,$haettava)
{
    for($x=0;$x<count($vertailu);$x++)
    {
        if ($vertailu[$x]== trim($Hakutieto[$haettava]))
        {
            $stempi=1;
        }
    }
    if ($stempi==0)
    {
        $vertailu[] = trim($Hakutieto[$haettava]);
    }
    $stempi=0;
}

function Lista($lista,$valittu)
{
    for( $x=0;$x<count($lista);$x++)
    {
        //if ($valittu == "") $valittu = $lista[0];

        if ($valittu == $lista[$x]) echo "<option selected>$lista[$x]</option>";
        else echo "<option> $lista[$x] </option>";
    }
}

function Jarjestely($Taulukko,$Styyli,$lisatyyli,$stosilisatyyli)
{
    if ($Styyli == Pisteet || $Styyli==YVPisteet || $Styyli== AVPisteet)
    {
        for($t=0;$t<count($Taulukko);$t++)
        for($x=1;$x<count($Taulukko);$x++)
        {
            if ($Taulukko[$x-1][$Styyli] < $Taulukko[$x][$Styyli] )
            {
                $temp =$Taulukko[$x];
                $Taulukko[$x]=$Taulukko[$x-1];
                $Taulukko[$x-1]=$temp;
            }
            if ($Taulukko[$x-1][$Styyli] == $Taulukko[$x][$Styyli] && $Taulukko[$x-1][$lisatyyli] <
$Taulukko[$x][$lisatyyli])
            {
                $temp =$Taulukko[$x];
                $Taulukko[$x]=$Taulukko[$x-1];
                $Taulukko[$x-1]=$temp;
            }
            if ($Taulukko[$x-1][$Styyli] == $Taulukko[$x][$Styyli] && $Taulukko[$x-1][$lisatyyli] <=
$Taulukko[$x][$lisatyyli] && $Taulukko[$x-1][$stosilisatyyli] < $Taulukko[$x][$stosilisatyyli])
            {
                $temp =$Taulukko[$x];
                $Taulukko[$x]=$Taulukko[$x-1];
                $Taulukko[$x-1]=$temp;
            }
        }
    }
}

function isoLista($isoLista,$valittu)
{
    for( $x=0;$x<count($isoLista);$x++)
    {
        if ($valittu[0] == "") $valittu[0] = "Kaikki";
        for( $y=0;$y<count($valittu);$y++)
        if( $valittu[$y] == $isoLista[$x]) $Samat =1;
        if ($Samat == 1) echo "<option selected>$isoLista[$x]</option>";
        else echo "<option> $isoLista[$x] </option>";
        $Samat =0;
    }
}

function Viiminen($shaluttu,$sana)
{
    for($x=count($shaluttu);$x>0;$x--)
    $shaluttu[$x]=$shaluttu[($x-1)];
}

```


Liite 1 4(26)

```

        $shaluttu[0] = $sana;
    }
    $sosote = mysql_connect( $host , $skayttaja , $salasana );

    if ( $pknimi == "" && $pkjoukkue == "" )
    {
        //sarjataso ja lohko valinnat
        $kysely = "SELECT * FROM ottelut;";

        if ( ! $sosote ) die( "Tietokantaan ei saatu yhteyttä!" );
        mysql_select_db( $tietokanta, $sosote ) or die ( "Tietokanta $tietokanta: ." ei saatu valittua".mysql_error() );
        $sqlhakutieto = mysql_query( $kysely, $sosote );
        while ( $sqlrivi = mysql_fetch_array( $sqlhakutieto ) )
        {
            Taulukointi(&$shaku_sarjatasot,$sqlrivi,"Sarjataso");
        }
        if ( $Sarjataso_valinta == "" ) $SarjatasoHaku_valinta = $shaku_sarjatasot[0];
        else $SarjatasoHaku_valinta = $Sarjataso_valinta;
        $kysely = "SELECT * FROM ottelut WHERE Sarjataso ='$SarjatasoHaku_valinta';";
        $sqlhakutieto = mysql_query( $kysely, $sosote );
        while ( $sqlrivi = mysql_fetch_array( $sqlhakutieto ) )
        {
            Taulukointi(&$shaku_lohkot,$sqlrivi,"Lohko");
        }
        $kysely = "SELECT * FROM ottelut WHERE Sarjataso ='$SarjatasoHaku_valinta' ";
        $kysely .=" AND ( ";
        for($x=0;$x<count($shaku_lohkot);$x++)
        {
            $kysely .=" Lohko = '$shaku_lohkot[$x]' ";
            if(count($shaku_lohkot) >=2 && $x<(count($shaku_lohkot)-1)) $kysely .=" OR ";
        }
        $kysely .=")";
        //tähän lopppuu sarjaso ja lohko kysely

        $sqlhakutieto = mysql_query( $kysely, $sosote );
        while ( $sqlrivi = mysql_fetch_array( $sqlhakutieto ) )
        {
            Taulukointi(&$shaku_joukkueet,$sqlrivi,"Kotijoukkue");
            Taulukointi(&$shaku_joukkueet,$sqlrivi,"Vierasjoukkue");
            Taulukointi(&$shaku_kaudet,$sqlrivi,"Kausi");
            Taulukointi(&$shaku_ottelut,$sqlrivi,"Ottelu");
        }

        /*
        $pelaajakysely = "SELECT * FROM pelaajatiedot WHERE Sarjataso ='$SarjatasoHaku_valinta' AND
        Lohko='$LohkoHaku_valinta';";
        if ( $LohkoHaku_valinta == "Kaikki" ) $pelaajakysely = "SELECT * FROM pelaajatiedot WHERE Sarjataso
        ='$SarjatasoHaku_valinta'; ";
        $sqlhakutieto = mysql_query( $pelaajakysely, $sosote );
        while ( $sqlrivi = mysql_fetch_array( $sqlhakutieto ) )
        {
            // Taulukointi(&$shaku_pelaajat,$sqlrivi,"Nimi");
        }
        */

        if ( count($Kausi_valinta) == 1 && $Kausi_valinta[0] != "Kaikki" ) $soorait1 =1;
        if ( count($Kausi_valinta) >1 ) $soorait1 =1;
        if ( count($Ottelu_valinta) == 1 && $Ottelu_valinta[0] != "Kaikki" ) $soorait2 =1;
        if ( count($Ottelu_valinta) >1 ) $soorait2 =1;
        if ( count($Joukkue_valinta) == 1 && $Joukkue_valinta[0] != "Kaikki" ) $soorait3 =1;
        if ( count($Joukkue_valinta) >1 ) $soorait3 =1;
        if ( count($Lohko_valinta) == 1 && $Lohko_valinta[0] != "Kaikki" ) $soorait4 =1;
        if ( count($Lohko_valinta) >1 ) $soorait4 =1;

        if ( ($soorait1 == 1 || $soorait2 == 1 || $soorait3 == 1 || $soorait4 == 1 || $Pelaaja_valinta != "Kaikki" ) && isset($Tilasto_valinta) )
        {
            $lisakysely = "SELECT * FROM ottelut WHERE Sarjataso = '$Sarjataso_valinta' ";
            if ( count($Lohko_valinta)>0 )
            {
                if ( count($Lohko_valinta)==1 && $Lohko_valinta[0] == "Kaikki" ) $Lohkot = "Kaikki";
                else
                {
                    if( $Lohko_valinta[0] == "Kaikki" )
                    {
                        $lisakysely .=" AND ( ";
                        for($x=1;$x<count($Lohko_valinta);$x++)
                        {

```

Liite 1 5(26)

```

        $lisakysely .=" Lohko = '$Lohko_valinta[$x]' ";
        if(count($Lohko_valinta) >2 && $x<(count($Lohko_valinta)-1)) $lisakysely .=" OR ";
    }
}
else
{
    $lisakysely .=" AND ( ";
    for($x=0;$x<count($Lohko_valinta);$x++)
    {
        $lisakysely .=" Lohko = '$Lohko_valinta[$x]' ";
        if(count($Lohko_valinta) >=2 && $x<(count($Lohko_valinta)-1)) $lisakysely .=" OR ";
    }
    $lisakysely .=")";
}
}
if (count($Kausi_valinta)>0)
{
    if (count($Kausi_valinta)==1 && $Kausi_valinta[0] == "Kaikki") $Kaudet = "Kaikki";
    else
    {
        if( $Kausi_valinta[0] == "Kaikki")
        {
            $lisakysely .=" AND ( ";
            for($x=1;$x<count($Kausi_valinta);$x++)
            {
                $lisakysely .=" Kausi = '$Kausi_valinta[$x]' ";
                if(count($Kausi_valinta) >2 && $x<(count($Kausi_valinta)-1)) $lisakysely .=" OR ";
            }
        }
        else
        {
            $lisakysely .=" AND ( ";
            for($x=0;$x<count($Kausi_valinta);$x++)
            {
                $lisakysely .=" Kausi = '$Kausi_valinta[$x]' ";
                if(count($Kausi_valinta) >=2 && $x<(count($Kausi_valinta)-1)) $lisakysely .=" OR ";
            }
            $lisakysely .=")";
        }
    }
}
if (count($Ottelu_valinta)>0)
{
    if (count($Ottelu_valinta)==1 && $Ottelu_valinta[0] == "Kaikki") $Ottelut = "Kaikki";
    else
    {
        if( $Ottelu_valinta[0] == "Kaikki")
        {
            $lisakysely .=" AND ( ";
            for($x=1;$x<count($Ottelu_valinta);$x++)
            {
                $lisakysely .=" Ottelu = '$Ottelu_valinta[$x]' ";
                if(count($Ottelu_valinta) >2 && $x<(count($Ottelu_valinta)-1)) $lisakysely .=" OR ";
            }
        }
        else
        {
            $lisakysely .=" AND ( ";
            for($x=0;$x<count($Ottelu_valinta);$x++)
            {
                $lisakysely .=" Ottelu = '$Ottelu_valinta[$x]' ";
                if(count($Ottelu_valinta) >=2 && $x<(count($Ottelu_valinta)-1)) $lisakysely .=" OR ";
            }
            $lisakysely .=")";
        }
    }
}
if (count($Joukkue_valinta)>0 )
{
    /*
    if (count($Joukkue_valinta)==1 && $Joukkue_valinta[0] == "Kaikki") $Joukkueet = "Kaikki";
    else
    {

```

Liite 1 6(26)

```

if (count($Joukkue_valinta)==1 && $Joukkue_valinta[0] != "Kaikki")
{
    $lisakysely .=" AND Kotijoukkue = '$Joukkue_valinta[0]' OR Vierasjoukkue =
'$Joukkue_valinta[0]';
}
if (count($Joukkue_valinta)==2 && $Joukkue_valinta[0] == "Kaikki")
{
    $lisakysely .=" AND Kotijoukkue = '$Joukkue_valinta[1]' OR Vierasjoukkue = '
'$Joukkue_valinta[1]';
}
else
{
    $lisakysely .=" AND ( Kotijoukkue = '$Joukkue_valinta[0]' AND Vierasjoukkue =
'$Joukkue_valinta[1]' ";
    $lisakysely .=" OR Kotijoukkue = '$Joukkue_valinta[1]' AND Vierasjoukkue =
'$Joukkue_valinta[0]' ";
}
}
*/
if (count($Joukkue_valinta)==1 && $Joukkue_valinta[0] == "Kaikki") $Joukkueet = "Kaikki";
else
{
    if( $Joukkue_valinta[0] == "Kaikki")
    {
        $lisakysely .=" AND ( ";
        for($x=1;$x<count($Joukkue_valinta);$x++)
        {
            $lisakysely .=" Kotijoukkue = '$Joukkue_valinta[$x]' ";
            if(count($Joukkue_valinta) >2 && $x<(count($Joukkue_valinta)-1)) $lisakysely .="
OR ";
        }
    }
    else
    {
        $lisakysely .=" AND ( ";
        for($x=0;$x<count($Joukkue_valinta);$x++)
        {
            $lisakysely .=" Kotijoukkue = '$Joukkue_valinta[$x]' ";
            if(count($Joukkue_valinta) >=2 && $x<(count($Joukkue_valinta)-1)) $lisakysely .="
OR ";
        }
    }
    //$lisakysely .=")";
}
if (count($Joukkue_valinta)==1 && $Joukkue_valinta[0] == "Kaikki") $Joukkueet = "Kaikki";
else
{
    if( $Joukkue_valinta[0] == "Kaikki")
    {
        $lisakysely .=" OR ";
        for($x=1;$x<count($Joukkue_valinta);$x++)
        {
            $lisakysely .=" Vierasjoukkue = '$Joukkue_valinta[$x]' ";
            if(count($Joukkue_valinta) >2 && $x<(count($Joukkue_valinta)-1)) $lisakysely .="
OR ";
        }
    }
    else
    {
        $lisakysely .=" OR ";
        for($x=0;$x<count($Joukkue_valinta);$x++)
        {
            $lisakysely .=" Vierasjoukkue = '$Joukkue_valinta[$x]' ";
            if(count($Joukkue_valinta) >=2 && $x<(count($Joukkue_valinta)-1)) $lisakysely .="
OR ";
        }
    }
    $lisakysely .=")";
}
}

$lisakyselyottelu = $lisakysely;
$lisakyselyottelu .=",";
//echo $lisakyselyottelu;
if ( ! $sote ) die( "Tietokantaan ei saatu yhteyttä!" );
mysql_select_db( $tietokanta, $sote ) or die ( "Tietokantaa $tietokanta: "." ei saatu valittua.mysql_error() );

```

Liite 1 7(26)

```

$Sqlhakutieto = mysql_query( $Isakyselyottelu, $osote );
while ( $Sqlrivi = mysql_fetch_array( $Sqlhakutieto ) )
{
    Taulukointi(&$Isa_joukkue,$Sqlrivi,"Kotijoukkue");
    Taulukointi(&$Isa_joukkue,$Sqlrivi,"Vierasjoukkue");
    Taulukointi(&$Isa_kausi,$Sqlrivi,"Kausi");
    Taulukointi(&$Isa_ottelu,$Sqlrivi,"Ottelu");
    $Isapelaajatietoottelut[][ID]=$Sqlrivi[PeliID];
}
if (count($Isapelaajatietoottelut) !=0)
{
    $Isapelaajakysely = "SELECT * FROM pelaajatiedot ";
    if(count($Isapelaajatietoottelut) == 1) $Isapelaajakysely .= "WHERE PeliID="
    "". $Isapelaajatietoottelut[0][ID]."";
    else
    {
        $Isapelaajakysely .= "WHERE (";
        for($x=0;$x<count($Isapelaajatietoottelut);$x++)
        {
            $Isapelaajakysely .= " PeliID = " . $Isapelaajatietoottelut[$x][ID]."";
            if ($x != (count($Isapelaajatietoottelut)-1)) $Isapelaajakysely .= " OR ";
        }
        $Isapelaajakysely .= ")";
    }

    if (count($Joukkue_valinta)>0)
    {
        if (count($Joukkue_valinta)==1 && $Joukkue_valinta[0] == "Kaikki") $Joukkueet = "Kaikki";
        else
        {
            if( $Joukkue_valinta[0] == "Kaikki")
            {
                $Isapelaajakysely .= " AND ( ";
                for($x=1;$x<count($Joukkue_valinta);$x++)
                {
                    $Isapelaajakysely .= " Joukkue = '$Joukkue_valinta[$x]' ";
                    if(count($Joukkue_valinta) >2 && $x<(count($Joukkue_valinta)-1))
                    $Isapelaajakysely .= " OR ";
                }
            }
            else
            {
                $Isapelaajakysely .= " AND ( ";
                for($x=0;$x<count($Joukkue_valinta);$x++)
                {
                    $Isapelaajakysely .= " Joukkue = '$Joukkue_valinta[$x]' ";
                    if(count($Joukkue_valinta) >=2 && $x<(count($Joukkue_valinta)-1))
                    $Isapelaajakysely .= " OR ";
                }
            }
            $Isapelaajakysely .= ")";
        }
    }
    $Isapelaajakysely .= ";";

    // $osote = mysql_connect( $localhost , $skayttaja );
    if ( ! $osote ) die( "Tietokantaan ei saatu yhteyttä!" );
    mysql_select_db( $tietokanta, $osote ) or die ( "Tietokantaa $tietokanta: " . " ei saatu valittua".mysql_error() );
    $Sqlhakutieto = mysql_query( $Isapelaajakysely, $osote );
    while ( $Sqlrivi = mysql_fetch_array( $Sqlhakutieto ) )
    {
        Taulukointi(&$Isa_pelaajat,$Sqlrivi,"Nimi");
    }
}

//Valinnat aakkosjärjestykseen paitsi pelaajat joita ei tällä hetkellä käytetä
sort($shaku_sarjatasot);
sort($shaku_lohkot);
viiminen(&$shaku_lohkot,"Kaikki");
rsort($shaku_kaudet);
sort($shaku_ottelut);
viiminen(&$shaku_ottelut,"Kaikki");
sort($shaku_joukkueet);
viiminen(&$shaku_joukkueet,"Kaikki");
/*

```

Liite 1 8(26)

```

sort($shaku_pelaajat);
Viiminen(&$shaku_pelaajat,"Kaikki");
*/
rsort($lisa_kausi);
Viiminen(&$lisa_kausi,"Ei valittu");
sort($lisa_ottelu);
Viiminen(&$lisa_ottelu,"Ei valittu");
sort($lisa_joukkue);
Viiminen(&$lisa_joukkue,"Ei valittu");
sort($lisa_pelaajat);
Viiminen(&$lisa_pelaajat,"Ei valittu");
?>
<A NAME = "Alku"></A>
<form action="<?php $PHP_SELF ?>" method="<?php GET ?>">
<TABLE RULES=NONE FRAME=BOX class="nimet" align="center" scrolling="no" border="1" cellpadding="1" cellspacing="1"
width="800">
<TR><TD align="left" ><a target="_blank" href="pikaohje.txt">Pika ohjeet</a></TD>
<TD align="center" colspan="3"><a href="mailto:miikka.tikka@eng.tamk.fi">Palaute</a></TD>
<TD align="right" ><a target="_blank" href="/admin/admin.php">Adminvalikoihin</a></TD></TR>
<TR><TD class="otsikko" colspan="5"> Salibandytilastot</TD></TR>

<TR><TD colspan="2">Tilastojenvalinta:<select class="valinnat" name="Tilasto_valinta">
<?php Lista($tilastovalinta,$Tilasto_valinta); ?>
</select></TD><TD colspan="2">Sarjataso:<select class="valinnat" name="Sarjataso_valinta">
<?php Lista($shaku_sarjatasot,$Sarjataso_valinta); ?>
</select></TD><TD><a target="_self" href="http://www.sctamko.com/">www.sctamko.com</a></TD></TR>
<TR><TD>Lohkot:</TD><TD>Kaudet:</TD><TD>Ottelut:</TD><TD>Joukkueet:</TD><TD align="middle"
rowspan="4"><input class="nappi" type="submit" value="OK!"></TD></TR>
<TR><TD><select class="valinnat" multiple size="4" name="Lohko_valinta[]">
<?php isoLista($shaku_lohkot,$Lohko_valinta); ?>
</select></TD><TD><select class="valinnat" multiple size="4" name="Kausi_valinta[]">
<?php isoLista($shaku_kaudet,$Kausi_valinta); ?>
</select></TD><TD>
<select class="valinnat" multiple size="4" name="Ottelu_valinta[]">;
<?php isoLista($shaku_ottelut,$Ottelu_valinta); ?>
</select></TD><TD>
<select class="valinnat" multiple size="4" name="Joukkue_valinta[]">
<?php isoLista($shaku_joukkueet,$Joukkue_valinta); ?>
</select></TD></TR>
<TR><TD rowspan="4">Pelaajatilastojen<BR>Valintoja:</TD><TD><select class="valinnat" name="Lisa_Kausi_valinta">;
<?php Lista($lisa_kausi,$Lisa_Kausi_valinta); ?>
</select></TD><TD>
<select class="valinnat" name="Lisa_Ottelu_valinta">
<?php Lista($lisa_ottelu,$Lisa_Ottelu_valinta); ?>
</select></TD><TD>
<select class="valinnat" name="Lisa_Joukkue_valinta">
<?php Lista($lisa_joukkue,$Lisa_Joukkue_valinta); ?>
</select></TD></TR><TR>
<TD>P&#246;rsit:</TD><TD>Lajittelu:</TD><TD>Pelaajat:<select class="valinnat" name="Pelaaja_valinta">
<?php Lista($shaku_pelaajat,$Pelaaja_valinta); ?>
</select></TD></TR><TR><TD>
<select class="valinnat" name="Porssi_valinta">
<?php Lista($shaku_porssit,$Porssi_valinta); ?>
</select></TD><TD>
<select class="valinnat" name="Haku_valinta">
<?php Lista($shaku_lajittelu,$Haku_valinta); ?>
</select></TD><TD>
<select class="valinnat" name="Lisa_Pelaaja_valinta">
<?php Lista($lisa_pelaajat,$Lisa_Pelaaja_valinta); ?>
</select></TD></TR><TR><TD rowspan="2">
<a href="Tilastot.php"
onMouseOver="handleOver();return true;"
onMouseOut="handleOut();return true;">
</a>
</TD><TD rowspan="2"></TR><TR>
<?php
if ($YVPinnat == 1) echo "<TD >YV pisteet: <input type='checkbox' name='YVPinnat' value='1' checked></TD>";
else echo "<TD >YV Pisteet: <input type='checkbox' name='YVPinnat' value='1' ></TD>";
if ($AVPinnat == 1) echo "<TD >AV pisteet: <input type='checkbox' name='AVPinnat' value='1' checked></TD>";
else echo "<TD >AV Pisteet: <input type='checkbox' name='AVPinnat' value='1' ></TD>";
if ($RLPinnat == 1) echo "<TD >RL pisteet: <input type='checkbox' name='RLPinnat' value='1' checked></TD>";
else echo "<TD >RL Pisteet: <input type='checkbox' name='RLPinnat' value='1' ></TD>";
echo "</TR></TABLE></form>";

if (isset($tilastovalinta))
{

```

Liite 1 9(26)

```

Skysely = " WHERE Sarjataso = '$Sarjataso_valinta' ";
if (count($Lohko_valinta)>0)
{
    if (count($Lohko_valinta)==1 && $Lohko_valinta[0] == "Kaikki") $Lohkot = "Kaikki";
    else
    {
        if( $Lohko_valinta[0] == "Kaikki")
        {
            $skysely .= " AND ( ";
            for($x=1;$x<count($Lohko_valinta);$x++)
            {
                $skysely .= " Lohko = '$Lohko_valinta[$x]' ";
                if(count($Lohko_valinta) >2 && $x<(count($Lohko_valinta)-1)) $skysely .= " OR ";
            }
        }
        else
        {
            $skysely .= " AND ( ";
            for($x=0;$x<count($Lohko_valinta);$x++)
            {
                $skysely .= " Lohko = '$Lohko_valinta[$x]' ";
                if(count($Lohko_valinta) >=2 && $x<(count($Lohko_valinta)-1)) $skysely .= " OR ";
            }
        }
        $skysely .= " )";
    }
}
if (count($Kausi_valinta)>0)
{
    if (count($Kausi_valinta)==1 && $Kausi_valinta[0] == "Kaikki") $Kaudet = "Kaikki";
    else
    {
        if( $Kausi_valinta[0] == "Kaikki")
        {
            $skysely .= " AND ( ";
            for($x=1;$x<count($Kausi_valinta);$x++)
            {
                $skysely .= " Kausi = '$Kausi_valinta[$x]' ";
                if(count($Kausi_valinta) >2 && $x<(count($Kausi_valinta)-1)) $skysely .= " OR ";
            }
        }
        else
        {
            $skysely .= " AND ( ";
            for($x=0;$x<count($Kausi_valinta);$x++)
            {
                $skysely .= " Kausi = '$Kausi_valinta[$x]' ";
                if(count($Kausi_valinta) >=2 && $x<(count($Kausi_valinta)-1)) $skysely .= " OR ";
            }
        }
        $skysely .= " )";
    }
}
if (count($Ottelu_valinta)>0)
{
    if (count($Ottelu_valinta)==1 && $Ottelu_valinta[0] == "Kaikki") $Ottelut = "Kaikki";
    else
    {
        if( $Ottelu_valinta[0] == "Kaikki")
        {
            $skysely .= " AND ( ";
            for($x=1;$x<count($Ottelu_valinta);$x++)
            {
                $skysely .= " Ottelu = '$Ottelu_valinta[$x]' ";
                if(count($Ottelu_valinta) >2 && $x<(count($Ottelu_valinta)-1)) $skysely .= " OR ";
            }
        }
        else
        {
            $skysely .= " AND ( ";
            for($x=0;$x<count($Ottelu_valinta);$x++)
            {
                $skysely .= " Ottelu = '$Ottelu_valinta[$x]' ";
                if(count($Ottelu_valinta) >=2 && $x<(count($Ottelu_valinta)-1)) $skysely .= " OR ";
            }
        }
        $skysely .= " )";
    }
}

```

Liite 1 10(26)

```

    }
}

if (count($Joukkue_valinta)>0)
{
    if (count($Joukkue_valinta)==1 && $Joukkue_valinta[0] == "Kaikki") $Joukkueet = "Kaikki";
    else
    {
        if( $Joukkue_valinta[0] == "Kaikki")
        {
            $kysely .=" AND ( ";
            for($x=1;$x<count($Joukkue_valinta);$x++)
            {
                $kysely .=" Kotijoukkue = '$Joukkue_valinta[$x]' ";
                if(count($Joukkue_valinta) >2 && $x<(count($Joukkue_valinta)-1)) $kysely .=" OR ";
            }
        }
        else
        {
            $kysely .=" AND ( ";
            for($x=0;$x<count($Joukkue_valinta);$x++)
            {
                $kysely .=" Kotijoukkue = '$Joukkue_valinta[$x]' ";
                if(count($Joukkue_valinta) >=2 && $x<(count($Joukkue_valinta)-1)) $kysely .=" OR ";
            }
        }
    }
}

if (count($Joukkue_valinta)>0)
{
    if (count($Joukkue_valinta)==1 && $Joukkue_valinta[0] == "Kaikki") $Joukkueet = "Kaikki";
    else
    {
        if( $Joukkue_valinta[0] == "Kaikki")
        {
            $kysely .=" OR ";
            for($x=1;$x<count($Joukkue_valinta);$x++)
            {
                $kysely .=" Vierasjoukkue = '$Joukkue_valinta[$x]' ";
                if(count($Joukkue_valinta) >2 && $x<(count($Joukkue_valinta)-1)) $kysely .=" OR ";
            }
        }
        else
        {
            $kysely .=" OR ";
            for($x=0;$x<count($Joukkue_valinta);$x++)
            {
                $kysely .=" Vierasjoukkue = '$Joukkue_valinta[$x]' ";
                if(count($Joukkue_valinta) >=2 && $x<(count($Joukkue_valinta)-1)) $kysely .=" OR ";
            }
        }
        $kysely .=")";
    }
}

//lisä haut
if ($Lohkot != "Kaikki" || $Kaudet != "Kaikki" || $Joukkueet != "Kaikki" || $Ottelut != "Kaikki" || $Pelaaja_valinta != "Kaikki")
{
    if ($Lisa_Joukkue_valinta != "Ei valittu" || $Lisa_Kausi_valinta != "Ei valittu" || $Lisa_Ottelu_valinta != "Ei valittu") $kysely .=" AND ";
    if ($Lisa_Joukkue_valinta != "Ei valittu") $kysely .=" Kotijoukkue = '$Lisa_Joukkue_valinta' OR Vierasjoukkue = '$Lisa_Joukkue_valinta' ";
    if ($Lisa_Joukkue_valinta != "Ei valittu" && $Lisa_Kausi_valinta != "Ei valittu") $kysely .=" AND ";
    if ($Lisa_Kausi_valinta != "Ei valittu") $kysely .=" Kausi = '$Lisa_Kausi_valinta' ";
    if (($Lisa_Joukkue_valinta != "Ei valittu" || $Lisa_Kausi_valinta != "Ei valittu") && $Lisa_Ottelu_valinta != "Ei valittu") $kysely .=" AND ";
    if ($Lisa_Ottelu_valinta != "Ei valittu") $kysely .=" Ottelu = '$Lisa_Ottelu_valinta' ";
}

}

if ($Tilasto_valinta == "Pelaajatilastot")
{

```

Liite 1 11(26)

```

$skysely = " WHERE Sarjataso = '$Sarjataso_valinta' ";
if (count($Lohko_valinta)>0)
{
    if (count($Lohko_valinta)==1 && $Lohko_valinta[0] == "Kaikki") $Lohkot = "Kaikki";
    else
    {
        if( $Lohko_valinta[0] == "Kaikki")
        {
            $skysely .= " AND ( ";
            for($x=1;$x<count($Lohko_valinta);$x++)
            {
                $skysely .= " Lohko = '$Lohko_valinta[$x]' ";
                if(count($Lohko_valinta) >2 && $x<(count($Lohko_valinta)-1)) $skysely .= " OR ";
            }
        }
        else
        {
            $skysely .= " AND ( ";
            for($x=0;$x<count($Lohko_valinta);$x++)
            {
                $skysely .= " Lohko = '$Lohko_valinta[$x]' ";
                if(count($Lohko_valinta) >=2 && $x<(count($Lohko_valinta)-1)) $skysely .= " OR ";
            }
        }
        $skysely .=")";
    }
}
if (count($Kausi_valinta)>0)
{
    if (count($Kausi_valinta)==1 && $Kausi_valinta[0] == "Kaikki") $Kaudet = "Kaikki";
    else
    {
        if( $Kausi_valinta[0] == "Kaikki")
        {
            $skysely .= " AND ( ";
            for($x=1;$x<count($Kausi_valinta);$x++)
            {
                $skysely .= " Kausi = '$Kausi_valinta[$x]' ";
                if(count($Kausi_valinta) >2 && $x<(count($Kausi_valinta)-1)) $skysely .= " OR ";
            }
        }
        else
        {
            $skysely .= " AND ( ";
            for($x=0;$x<count($Kausi_valinta);$x++)
            {
                $skysely .= " Kausi = '$Kausi_valinta[$x]' ";
                if(count($Kausi_valinta) >=2 && $x<(count($Kausi_valinta)-1)) $skysely .= " OR ";
            }
        }
        $skysely .=")";
    }
}
if (count($Ottelu_valinta)>0)
{
    if (count($Ottelu_valinta)==1 && $Ottelu_valinta[0] == "Kaikki") $Ottelut = "Kaikki";
    else
    {
        if( $Ottelu_valinta[0] == "Kaikki")
        {
            $skysely .= " AND ( ";
            for($x=1;$x<count($Ottelu_valinta);$x++)
            {
                $skysely .= " Ottelu = '$Ottelu_valinta[$x]' ";
                if(count($Ottelu_valinta) >2 && $x<(count($Ottelu_valinta)-1)) $skysely .= " OR ";
            }
        }
        else
        {
            $skysely .= " AND ( ";
            for($x=0;$x<count($Ottelu_valinta);$x++)
            {
                $skysely .= " Ottelu = '$Ottelu_valinta[$x]' ";
                if(count($Ottelu_valinta) >=2 && $x<(count($Ottelu_valinta)-1)) $skysely .= " OR ";
            }
        }
        $skysely .=")";
    }
}

```


Liite 1 12(26)

```

    }
}
if (count($Joukkue_valinta)>0)
{
    if (count($Joukkue_valinta)==1 && $Joukkue_valinta[0] == "Kaikki") $Joukkueet = "Kaikki";
    else
    {
        if ($Joukkue_valinta[0] == "Kaikki")
        {
            $kysely .=" AND ( ";
            for($x=1;$x<count($Joukkue_valinta);$x++)
            {
                $kysely .=" Joukkue = '$Joukkue_valinta[$x]' ";
                if(count($Joukkue_valinta) >2 && $x<(count($Joukkue_valinta)-1)) $kysely .=" OR ";
            }
        }
        else
        {
            $kysely .=" AND ( ";
            for($x=0;$x<count($Joukkue_valinta);$x++)
            {
                $kysely .=" Joukkue = '$Joukkue_valinta[$x]' ";
                if(count($Joukkue_valinta) >=2 && $x<(count($Joukkue_valinta)-1)) $kysely .=" OR ";
            }
            $kysely .=")";
        }
    }
}

//lisä haut
if ($Lohkot != "Kaikki" || $Kaudet != "Kaikki" || $Joukkueet != "Kaikki" || $Ottelut != "Kaikki" || $Pelaaja_valinta != "Kaikki")
{
    if ($Lisa_Joukkue_valinta != "Ei valittu" || $Lisa_Kausi_valinta != "Ei valittu" || $Lisa_Ottelu_valinta != "Ei valittu" ) $kysely .=" AND ";
    if ($Lisa_Joukkue_valinta != "Ei valittu") $kysely .=" Joukkue = '$Lisa_Joukkue_valinta' ";
    if ($Lisa_Joukkue_valinta != "Ei valittu" && $Lisa_Kausi_valinta != "Ei valittu") $kysely .=" AND ";
    if ($Lisa_Kausi_valinta != "Ei valittu") $kysely .=" Kausi = '$Lisa_Kausi_valinta' ";
    if (($Lisa_Joukkue_valinta != "Ei valittu" || $Lisa_Kausi_valinta != "Ei valittu") && $Lisa_Ottelu_valinta != "Ei valittu") $kysely .=" AND ";
    if ($Lisa_Ottelu_valinta != "Ei valittu" ) $kysely .=" Ottelu= '$Lisa_Ottelu_valinta' ";
}

if ($Pelaaja_valinta != "Kaikki") $kysely .=" AND Nimi = '$Pelaaja_valinta'";
//lisä haku
if ( $Lohkot != "Kaikki" || $Kaudet != "Kaikki" || $Joukkueet != "Kaikki" || $Ottelut != "Kaikki" || $Pelaaja_valinta != "Kaikki")
{
    if ($Lisa_Pelaaja_valinta != "Ei valittu") $kysely .=" AND Nimi = '$Lisa_Pelaaja_valinta'";
}
if ( $Haku_valinta != "Ei mitään") $kysely .=" ORDER BY $Haku_valinta DESC";

$kysely .="";
$Pelaajatietokysely = "SELECT * FROM pelaajatiedot ";
$Pelaajatietokysely .=$kysely;
//echo $Haku_valinta."<br>";
//echo $Pelaajatietokysely;
//$osote = mysql_connect( $host , $kayttaja );
//if ( ! $osote ) die( "Tietokantaan ei saatu yhteyttä!" );
mysql_select_db( $tietokanta, $osote ) or die ( "Tietokantaa $tietokanta: "." ei saatu valittua".mysql_error() );
$sqlhakutieto = mysql_query( $Pelaajatietokysely, $osote );
while ( $sqlrivi = mysql_fetch_array( $sqlhakutieto ) )
{
    $Pelaajatietoottelut[][ID] = $sqlrivi[PeliID];
}
if (count($Pelaajatietoottelut) != 0)
{

//$osote = mysql_connect( $localhost , $kayttaja );
//if ( ! $osote ) die( "Tietokantaan ei saatu yhteyttä!" );
mysql_select_db( $tietokanta, $osote ) or die ( "Tietokantaa $tietokanta: "." ei saatu valittua".mysql_error() );
$sqlhakutieto = mysql_query( $Pelaajatietokysely, $osote );

echo "<BR>";
if ( $Porssi_valinta == "Pistepörssi") $nakyvat_pinnat="Pisteet";

```

Liite 1 13(26)

```

if ( $Porssi_valinta == "Maalipörssi" ) $nakyvat_pinnat="Maalit";
if ( $Porssi_valinta == "Syöttöpörssi" ) $nakyvat_pinnat="Syotot";
if ( $Porssi_valinta == "Jäähypörssi" ) $nakyvat_pinnat="Jaahyt";
if ( $Porssi_valinta == "YVMaalit" ) $nakyvat_pinnat="YVMaalit";
if ( $Porssi_valinta == "YVSYötöt" ) $nakyvat_pinnat="YVSYotot";
if ( $Porssi_valinta == "YVPisteet" ) $nakyvat_pinnat="YVPisteet";
if ( $Porssi_valinta == "AVMaalit" ) $nakyvat_pinnat="AVMaalit";
if ( $Porssi_valinta == "AVSYötöt" ) $nakyvat_pinnat="AVSYotot";
if ( $Porssi_valinta == "AVPisteet" ) $nakyvat_pinnat="AVPisteet";
if ( $Porssi_valinta == "RLYritykset" ) $nakyvat_pinnat="RLYritykset";
if ( $Porssi_valinta == "RLMaalit" ) $nakyvat_pinnat="RLMaalit";

if ( $Porssi_valinta != "Ei Mitään" )
{
    //Haetaan ne pelaajat joilla on halutun pörssivalinnan kohdalla enemmän kuin "0":lla
    while ( $sqlrivi = mysql_fetch_array( $sqlhakutieto ) )
    {
        if ( $sqlrivi[Toimihenkilö] == "0" && $sqlrivi[$nakyvat_pinnat] > "0" )
        {
            for( $x=0; $x<count($porssi_pelaajat); $x++ )
            {
                if ( $porssi_pelaajat[$x][Nimi] == $sqlrivi[Nimi] )
                {
                    Stemppi=1;
                }
            }
            if ( $Stemppi==0 )
            {
                //Strimmaus = trim($sqlrivi[Nimi]);
                $porssi_pelaajat[$x][Nimi] = $sqlrivi[Nimi];
            }
            $Stemppi=0;
        }
    }
    //Lasketaan tilastot edellisessä silmukassa haetuille pelaajille
    $sqlhakutieto = mysql_query( $pelaajatietokysely, $osote );
    while ( $sqlrivi = mysql_fetch_array( $sqlhakutieto ) )
    {
        for( $x=0; $x<count($porssi_pelaajat); $x++ )
        {
            if ( $porssi_pelaajat[$x][Nimi] == $sqlrivi[Nimi] )
            {
                $porssi_pelaajat[$x][Maalit] += $sqlrivi[Maalit];
                $porssi_pelaajat[$x][Syotot] += $sqlrivi[Syotot];
                $porssi_pelaajat[$x][Pisteet] += $sqlrivi[Pisteet];
                $porssi_pelaajat[$x][Jaahyt] += $sqlrivi[Jaahyt];
                $porssi_pelaajat[$x][Ottelut] += 1;
                $porssi_pelaajat[$x][YVMaalit] += $sqlrivi[YVMaalit];
                $porssi_pelaajat[$x][YVSYotot] += $sqlrivi[YVSYotot];
                $porssi_pelaajat[$x][YVPisteet] += $sqlrivi[YVPisteet];
                $porssi_pelaajat[$x][AVMaalit] += $sqlrivi[AVMaalit];
                $porssi_pelaajat[$x][AVSYotot] += $sqlrivi[AVSYotot];
                $porssi_pelaajat[$x][AVPisteet] += $sqlrivi[AVPisteet];
                $porssi_pelaajat[$x][RLYritykset] += $sqlrivi[RLYritykset];
                $porssi_pelaajat[$x][RLMaalit] += $sqlrivi[RLMaalit];
                $porssi_pelaajat[$x][Joukkue] = $sqlrivi[Joukkue];
            }
        }
    }
    //järjestellään pelaajat suuruus järjestykseen funktiossa
    if ( $Porssi_valinta == "Pistepörssi" ) $porssi_pelaajat = SQLSort($porssi_pelaajat,"Pisteet","Maalit");
    if ( $Porssi_valinta == "Maalipörssi" ) $porssi_pelaajat = SQLSort($porssi_pelaajat,"Maalit","");
    if ( $Porssi_valinta == "Syöttöpörssi" ) $porssi_pelaajat = SQLSort($porssi_pelaajat,"Syotot","");
    if ( $Porssi_valinta == "Jäähypörssi" ) $porssi_pelaajat = SQLSort($porssi_pelaajat,"Jaahyt","Jaahyt");
    if ( $Porssi_valinta == "YVMaalit" ) $porssi_pelaajat = SQLSort($porssi_pelaajat,"YVMaalit","");
    if ( $Porssi_valinta == "YVSYötöt" ) $porssi_pelaajat = SQLSort($porssi_pelaajat,"YVSYotot","");
    if ( $Porssi_valinta == "YVPisteet" ) $porssi_pelaajat = SQLSort($porssi_pelaajat,"YVPisteet","YVMaalit");
    if ( $Porssi_valinta == "AVMaalit" ) $porssi_pelaajat = SQLSort($porssi_pelaajat,"AVMaalit","");
    if ( $Porssi_valinta == "AVSYötöt" ) $porssi_pelaajat = SQLSort($porssi_pelaajat,"AVSYotot","");
    if ( $Porssi_valinta == "AVPisteet" ) $porssi_pelaajat = SQLSort($porssi_pelaajat,"AVPisteet","AVMaalit");
    if ( $Porssi_valinta == "RLYritykset" ) $porssi_pelaajat = SQLSort($porssi_pelaajat,"RLYritykset","");
    if ( $Porssi_valinta == "RLMaalit" ) $porssi_pelaajat = SQLSort($porssi_pelaajat,"RLMaalit","");

    //Pörssien tulostukset
    ?>

```


Liite 1 15(26)

```

        echo "<TD>${sqlrivi[Pisteet]}</TD>";
        echo "<TD>${sqlrivi[Jaahyt]}</TD>";
        echo "<TD>${sqlrivi[YVMaalit]}</TD>";
        echo "<TD>${sqlrivi[YVSyotot]}</TD>";
        echo "<TD>${sqlrivi[YVPisteet]}</TD>";
        echo "<TD>${sqlrivi[AVMaalit]}</TD>";
        echo "<TD>${sqlrivi[AVSyotot]}</TD>";
        echo "<TD>${sqlrivi[AVPisteet]}</TD>";
        echo "<TD>${sqlrivi[RLYritykset]}</TD>";
        echo "<TD>${sqlrivi[RLMaalit]}</TD></TR>";
    }
}

echo "</TABLE>";

}

?>
<TABLE align="center" border="0" cellpadding="0" cellspacing="0" width="800">
<TR><TD><A style="color: #FF0000" href="#Alku">Ylös</A></TD></TR></TABLE>
<?php
}
else
{
    ?>
    <TABLE align="center" border="0" cellpadding="0" cellspacing="0" width="800">
    <TR><TD class="otteluvirhe">Pelej&#228; ei l&#246;ytyntyn valituilla hakuehdoilla!</TD></TR></TABLE>
    <?php
}
}

if ($Tilasto_valinta == "Ottelut")
{
    $kysely .=";";
    $sottelukysely = "SELECT * FROM ottelut ";
    $sottelukysely .= $kysely;
    //echo $sottelukysely;
    // $sote = mysql_connect( $localhost , $kayttaja );
    //if ( ! $sote ) die( "Tietokantaan ei saatu yhteyttä!" );
    mysql_select_db( $tietokanta, $sote ) or die ( "Tietokantaa $tietokanta: "." ei saatu valittua".mysql_error() );

    $sqlhakutieto_ottelu = mysql_query( $sottelukysely, $sote );
    while ( $sqlrivi_ottelu = mysql_fetch_array( $sqlhakutieto_ottelu ) )
    {
        $smatsit[] = array ( Kotijoukkue => utf8_decode( $sqlrivi_ottelu[Kotijoukkue] ), Vierasjoukkue =>
        utf8_decode( $sqlrivi_ottelu[Vierasjoukkue] ), Pelipaikka => utf8_decode( $sqlrivi_ottelu[Pelipaikka] ),
        Paivamaara => $sqlrivi_ottelu[Paivamaara], Alkamisaika => $sqlrivi_ottelu[Alkamisaika], Loppumisaika =>
        $sqlrivi_ottelu[Loppumisaika], Kotiera1 => $sqlrivi_ottelu[Kotiera1], Kotiera2 => $sqlrivi_ottelu[Kotiera2],
        Kotiera3 => $sqlrivi_ottelu[Kotiera3], Vierasera1 => $sqlrivi_ottelu[Vierasera1], Vierasera2 =>
        $sqlrivi_ottelu[Vierasera2], Vierasera3 => $sqlrivi_ottelu[Vierasera3], KotiJA => $sqlrivi_ottelu[KotiJA],
        KotiRL => $sqlrivi_ottelu[KotiRL], VierasJA => $sqlrivi_ottelu[VierasJA], VierasRL =>
        $sqlrivi_ottelu[VierasRL], Kotiaikalisa => $sqlrivi_ottelu[Kotiaikalisa], Vierasaikalisa =>
        $sqlrivi_ottelu[Vierasaikalisa], Voittaja => utf8_decode( $sqlrivi_ottelu[Voittaja] ), Lisatiedot =>
        utf8_decode( $sqlrivi_ottelu[Lisatiedot] ), Ottelu => utf8_decode( $sqlrivi_ottelu[Ottelu] ), Lopputulos =>
        $sqlrivi_ottelu[Lopputulos], Rangaistuksetkoti => $sqlrivi_ottelu[Rangaistuksetkoti], Rangaistuksetvieras =>
        $sqlrivi_ottelu[Rangaistuksetvieras], Kausi => utf8_decode( $sqlrivi_ottelu[Kausi] ), Peli aika =>
        $sqlrivi_ottelu[Peli aika], EraMaara => $sqlrivi_ottelu[EraMaara], Sarjataso =>
        utf8_decode( $sqlrivi_ottelu[Sarjataso] ), Lohko => utf8_decode( $sqlrivi_ottelu[Lohko] ), PeliID=>
        $sqlrivi_ottelu[PeliID], Tuomari1=> utf8_decode( $sqlrivi_ottelu[Tuomari1] ), Tuomari2=>
        utf8_decode( $sqlrivi_ottelu[Tuomari2] );
    }

    if(count($smatsit)>0)
    {
        if (count($smatsit)<4)
        {
            $maali jaahy_kysely = "SELECT * FROM jaahytmaalit ";
            if(count($smatsit) == 1) $maali jaahy_kysely .= "WHERE PeliID= '". $smatsit[0][PeliID]. "'";
            else
            {
                $maali jaahy_kysely .= "WHERE ";
                for($x=0;$x<count($smatsit);$x++)
                {
                    $maali jaahy_kysely .= " PeliID= '". $smatsit[$x][PeliID]. "'";
                    if ($x != (count($smatsit)-1)) $maali jaahy_kysely .= " OR ";
                }
            }
        }
    }
}

```

Liite 1 16(26)

```

$maalijaahy_kysely .= ";";
$spelaajakysely = "SELECT * FROM pelaajatiedot ";
if(count($matsit) == 1) $spelaajakysely .= "WHERE PeliID= ".$matsit[0][PeliID]."";
else
{
    $spelaajakysely .= "WHERE ";
    for($x=0;$x<count($matsit);$x++)
    {
        $spelaajakysely .= " PeliID= ".$matsit[$x][PeliID]."";
        if ($x != (count($matsit)-1)) $spelaajakysely .= " OR ";
    }
}
$spelaajakysely .= ";";

//echo $kysely;
$sqlhakutieto_jm = mysql_query( $maalijaahy_kysely, $osote );
while ($sqlrivi_jm = mysql_fetch_array( $sqlhakutieto_jm ))
{
    for($x=0;$x<count($matsit);$x++)
    {
        if ($matsit[$x][PeliID] == $sqlrivi_jm[PeliID] && $sqlrivi_jm[JaahyVaiMaali] == "M" )
        {
            $Maalitiedot[$x][] = array( Paivays=>
                $sqlrivi_jm[Paivamaara],Ottelu=>utf8_decode($sqlrivi_jm[Ottelu]), Joukkue
                =>utf8_decode($sqlrivi_jm[Joukkue]), Numero =>$sqlrivi_jm[MJN], Aika
                =>$sqlrivi_jm[MAJMin], Tekija =>$sqlrivi_jm[MTJSyy], Syottaja
                =>$sqlrivi_jm[MSJAn],Koodi =>$sqlrivi_jm[MKJPa], Era =>
                $sqlrivi_jm[MJera]);
        }
        if ($matsit[$x][PeliID] == $sqlrivi_jm[PeliID] && $sqlrivi_jm[JaahyVaiMaali] == "J")
        {
            $Jaahytiedot[$x][] = array( Paivays=>
                $sqlrivi_jm[Paivamaara],Ottelu=>utf8_decode($sqlrivi_jm[Ottelu]), Joukkue
                =>utf8_decode($sqlrivi_jm[Joukkue]), Numero =>$sqlrivi_jm[MJN], Kesto
                =>$sqlrivi_jm[MAJMin], Syy =>$sqlrivi_jm[MTJSyy], Alku aika
                =>$sqlrivi_jm[MSJAn],Loppuaika =>$sqlrivi_jm[MKJPa], Era =>
                $sqlrivi_jm[MJera]);
        }
    }
}

$sqlhakutieto = mysql_query( $spelaajakysely, $osote );
while ($sqlrivi = mysql_fetch_array( $sqlhakutieto ))
{
    for($z=0;$z<count($matsit);$z++)
    for($x=0;$x<count($Maalitiedot[$z]);$x++)
    {
        if ( $Maalitiedot[$z][$x][Joukkue] == $sqlrivi[Joukkue] &&
            $Maalitiedot[$z][$x][Tekija] == $sqlrivi[Numero] && $Maalitiedot[$z][$x][PeliID] ==
            $sqlrivi_jm[PeliID]) $Maalitiedot[$z][$x][Tekija] = utf8_decode($sqlrivi[Nimi]);
        if ( $Maalitiedot[$z][$x][Joukkue] == $sqlrivi[Joukkue] &&
            $Maalitiedot[$z][$x][Syottaja] == $sqlrivi[Numero] && $Maalitiedot[$z][$x][PeliID]
            == $sqlrivi_jm[PeliID]) $Maalitiedot[$z][$x][Syottaja] = utf8_decode($sqlrivi[Nimi]);
    }
    for($z=0;$z<count($matsit);$z++)
    for($x=0;$x<count($Jaahytiedot[$z]);$x++)
    if ( $Jaahytiedot[$z][$x][Joukkue] == $sqlrivi[Joukkue] && $Jaahytiedot[$z][$x][Numero] ==
        $sqlrivi[Numero] && $Jaahytiedot[$z][$x][Ottelu] == $sqlrivi[Ottelu] &&
        $Jaahytiedot[$z][$x][PeliID] == $sqlrivi_jm[PeliID]) $Jaahytiedot[$z][$x][Numero] =
        utf8_decode($sqlrivi[Nimi]);
}

for($z=0;$z<count($matsit);$z++)
{
    for($x=0;$x<count($Maalitiedot[$z]);$x++)
    {
        $maalijat = explode( ":", $Maalitiedot[$z][$x][Aika]);
        $Maalitiedot[$z][$x][MaalinaikaMinutti] = $maalijat[0];
        $Maalitiedot[$z][$x][MaalinaikaSekunti] = $maalijat[1];
    }

    for ($x=0;$x<count($Maalitiedot[$z]);$x++)
    for ($y=1;$y<count($Maalitiedot[$z]);$y++)
    {
        $EdellMin = $Maalitiedot[$z][$y-1][MaalinaikaMinutti];
        $TamaMin = $Maalitiedot[$z][$y][MaalinaikaMinutti];
    }
}

```

Liite 1 17(26)

```

$EdelSek = $Maalitiedot[$z][$y-1][MaalinaikaSekunti];
$TamaSek = $Maalitiedot[$z][$y][MaalinaikaSekunti];

if ($TamaMin < $EdellMin )
{
    $stemp = $Maalitiedot[$z][$y];
    $Maalitiedot[$z][$y] = $Maalitiedot[$z][$y-1];
    $Maalitiedot[$z][$y-1] = $stemp;
}

if ($TamaMin == $EdellMin && $TamaSek < $EdelSek )
{
    $stemp = $Maalitiedot[$z][$y];
    $Maalitiedot[$z][$y] = $Maalitiedot[$z][$y-1];
    $Maalitiedot[$z][$y-1] = $stemp;
}
}
}
for($z=0;$z<count($matsit);$z++)
{
    for($x=0;$x<count($Jaahytiedot[$z]);$x++)
    {
        $jaahyajat = explode( ":", $Jaahytiedot[$z][$x][Alkuaika]);
        $Jaahytiedot[$z][$x][AlkuaikaMinutti] = $jaahyajat[0];
        $Jaahytiedot[$z][$x][MaalinaikaSekunti] = $jaahyajat[1];
    }
    for ($x=0;$x<count($Jaahytiedot[$z]);$x++)
    for ($y=1;$y<count($Jaahytiedot[$z]);$y++)
    {
        $EdellMin = $Jaahytiedot[$z][$y-1][AlkuaikaMinutti];
        $TamaMin = $Jaahytiedot[$z][$y][AlkuaikaMinutti];
        $EdelSek = $Jaahytiedot[$z][$y-1][MaalinaikaSekunti];
        $TamaSek = $Jaahytiedot[$z][$y][MaalinaikaSekunti];

        if ($TamaMin < $EdellMin )
        {
            $stemp = $Jaahytiedot[$z][$y];
            $Jaahytiedot[$z][$y] = $Jaahytiedot[$z][$y-1];
            $Jaahytiedot[$z][$y-1] = $stemp;
        }

        if ($TamaMin == $EdellMin && $TamaSek < $EdelSek )
        {
            $stemp = $Jaahytiedot[$z][$y];
            $Jaahytiedot[$z][$y] = $Jaahytiedot[$z][$y-1];
            $Jaahytiedot[$z][$y-1] = $stemp;
        }
    }
}
$jaahysyy_kysely = "SELECT * FROM jaahykoodit";
$sqlhakutieto = mysql_query( $jaahysyy_kysely, $osote );
while ($sqlrivi = mysql_fetch_array( $sqlhakutieto ))
for($z=0;$z<count($matsit);$z++)
for($x=0;$x<count($Jaahytiedot[$z]);$x++)
if ( $Jaahytiedot[$z][$x][Syy] == $sqlrivi[Koodi] ) $Jaahytiedot[$z][$x][Syy] =
utf8_decode($sqlrivi[Selitys]);
}

if ( count($matsit)>3 && $apu == 1)
{
    $stemp=$seka;
    $maaliyaahy_kysely = "SELECT * FROM jaahytmaalit WHERE ";
    for($seka;$seka<$viiminen;$seka++)
    {
        $maaliyaahy_kysely .= " PeliID= '". $matsit[$seka][PeliID]. "'";
        if ($seka != ($viiminen-1)) $maaliyaahy_kysely .= " OR ";
    }
    $seka=$stemp;
    $maaliyaahy_kysely .= ";";
    $pelaajakysely = "SELECT * FROM pelaajatiedot WHERE ";
    for($seka;$seka<$viiminen;$seka++)
    {
        $pelaajakysely .= " PeliID= '". $matsit[$seka][PeliID]. "'";
        if ($seka != ($viiminen-1)) $pelaajakysely .= " OR ";
    }
    $pelaajakysely .= ";";
    //echo $maaliyaahy_kysely;
}

```

Liite 1 18(26)

```

Seka=$temp;
$Sqlhakutieto_jm = mysql_query( $maalijaahy_kysely, $osote );
while ( $sqlrivi_jm = mysql_fetch_array( $Sqlhakutieto_jm ) )
{
    for($seka;$seka<$viiminin;$seka++)
    {
        if ( $matsit[$seka][PeliID] == $sqlrivi_jm[PeliID] && $sqlrivi_jm[JaahyVaiMaali] == "M" )
        {
            $Maalitiedot[$seka][] = array( Paivays=>
            $sqlrivi_jm[Paivamaara],Ottelu=>utf8_decode($sqlrivi_jm[Ottelu]), Joukkue
            =>utf8_decode($sqlrivi_jm[Joukkue]), Numero =>$sqlrivi_jm[MJN], Aika
            =>$sqlrivi_jm[MAJMin], Tekija =>$sqlrivi_jm[MTJSyy], Syottaja
            =>$sqlrivi_jm[MSJAn],Koodi =>$sqlrivi_jm[MKJPa], Era => $sqlrivi_jm[MJera]);
        }
        if ( $matsit[$seka][PeliID] == $sqlrivi_jm[PeliID] && $sqlrivi_jm[JaahyVaiMaali] == "J" )
        {
            $Jaahytiedot[$seka][] = array( Paivays=>
            $sqlrivi_jm[Paivamaara],Ottelu=>utf8_decode($sqlrivi_jm[Ottelu]), Joukkue
            =>utf8_decode($sqlrivi_jm[Joukkue]), Numero =>$sqlrivi_jm[MJN], Kesto
            =>$sqlrivi_jm[MAJMin], Syy =>$sqlrivi_jm[MTJSyy], Alku aika
            =>$sqlrivi_jm[MSJAn],Loppuaika =>$sqlrivi_jm[MKJPa], Era =>
            $sqlrivi_jm[MJera]);
        }
    }
    $seka=$temp;
}
$seka=$temp;
$Sqlhakutieto = mysql_query( $pelaajakysely, $osote );
while ( $sqlrivi = mysql_fetch_array( $Sqlhakutieto ) )
{
    for($seka;$seka<$viiminin;$seka++)
    for($x=0;$x<count($Maalitiedot[$seka]);$x++)
    {
        if ( $Maalitiedot[$seka][$x][Joukkue] == $sqlrivi[Joukkue] && $Maalitiedot[$seka][$x][Tekija]
        == $sqlrivi[Numero] && $Maalitiedot[$seka][$x][PeliID] == $sqlrivi_jm[PeliID])
        $Maalitiedot[$seka][$x][Tekija] = utf8_decode($sqlrivi[Nimi]);
        if ( $Maalitiedot[$seka][$x][Joukkue] == $sqlrivi[Joukkue] && $Maalitiedot[$seka][$x][Syottaja]
        == $sqlrivi[Numero] && $Maalitiedot[$seka][$x][PeliID] == $sqlrivi_jm[PeliID])
        $Maalitiedot[$seka][$x][Syottaja] = utf8_decode($sqlrivi[Nimi]);
    }
    $seka=$temp;
    for($seka;$seka<$viiminin;$seka++)
    for($x=0;$x<count($Jaahytiedot[$seka]);$x++)
    {
        if ( $Jaahytiedot[$seka][$x][Joukkue] == $sqlrivi[Joukkue] && $Jaahytiedot[$seka][$x][Numero]
        == $sqlrivi[Numero] && $Jaahytiedot[$seka][$x][Ottelu] == $sqlrivi[Ottelu] &&
        $Jaahytiedot[$seka][$x][PeliID] == $sqlrivi_jm[PeliID]) $Jaahytiedot[$seka][$x][Numero] =
        utf8_decode($sqlrivi[Nimi]);
    }
    $seka=$temp;
}
$seka=$temp;
for($seka;$seka<$viiminin;$seka++)
{
    for($x=0;$x<count($Maalitiedot[$seka]);$x++)
    {
        $maalijat = explode( ":", $Maalitiedot[$seka][$x][Aika]);
        $Maalitiedot[$seka][$x][MaalinaikaMinutti] = $maalijat[0];
        $Maalitiedot[$seka][$x][MaalinaikaSekunti] = $maalijat[1];
    }

    for ($x=0;$x<count($Maalitiedot[$seka]);$x++)
    for ($y=1;$y<count($Maalitiedot[$seka]);$y++)
    {
        $EdellMin = $Maalitiedot[$seka][$y-1][MaalinaikaMinutti];
        $TamaMin = $Maalitiedot[$seka][$y][MaalinaikaMinutti];
        $EdelSek = $Maalitiedot[$seka][$y-1][MaalinaikaSekunti];
        $TamaSek = $Maalitiedot[$seka][$y][MaalinaikaSekunti];

        if ($TamaMin < $EdellMin )
        {
            $vara = $Maalitiedot[$seka][$y];
            $Maalitiedot[$seka][$y] = $Maalitiedot[$seka][$y-1];
            $Maalitiedot[$seka][$y-1] = $vara;
        }
    }
}

```

Liite 1 19(26)

```

        if ($TamaMin == $EdellMin && $TamaSek < $EdelSek )
        {
            $vara =$Maalitiedot[$seka][$y];
            $Maalitiedot[$seka][$y]=$Maalitiedot[$seka][$y-1];
            $Maalitiedot[$seka][$y-1]=$vara;
        }
    }
}
$seka=$temp;
for($seka;$seka<$viimininen;$seka++)
{
    for($x=0;$x<count($jaahytiedot[$z]);$x++)
    {
        $jaahyajat = explode( ":", $jaahytiedot[$seka][$x][Alkuaika]);
        $jaahytiedot[$seka][$x][AlkuaikaMinutti] = $jaahyajat[0];
        $jaahytiedot[$seka][$x][MaalinaikaSekunti] = $jaahyajat[1];
    }
    for ($x=0;$x<count($jaahytiedot[$seka]);$x++)
    for ($y=1;$y<count($jaahytiedot[$seka]);$y++)
    {
        $EdellMin = $jaahytiedot[$seka][$y-1][AlkuaikaMinutti];
        $TamaMin = $jaahytiedot[$seka][$y][AlkuaikaMinutti];
        $EdelSek = $jaahytiedot[$seka][$y-1][MaalinaikaSekunti];
        $TamaSek = $jaahytiedot[$seka][$y][MaalinaikaSekunti];

        if ($TamaMin < $EdellMin )
        {
            $vara =$jaahytiedot[$seka][$y];
            $jaahytiedot[$seka][$y]=$jaahytiedot[$seka][$y-1];
            $jaahytiedot[$seka][$y-1]=$vara;
        }

        if ($TamaMin == $EdellMin && $TamaSek < $EdelSek )
        {
            $vara =$jaahytiedot[$seka][$y];
            $jaahytiedot[$seka][$y]=$jaahytiedot[$seka][$y-1];
            $jaahytiedot[$seka][$y-1]=$vara;
        }
    }
}

$seka=$temp;
$jaahysyy_kysely = "SELECT * FROM jaahykoodit;";
$sqlhakutieto = mysql_query( $jaahysyy_kysely, $osote );
while ($sqlrivi = mysql_fetch_array( $sqlhakutieto ))
{
    for($seka;$seka<$viimininen;$seka++)
    for($x=0;$x<count($jaahytiedot[$seka]);$x++)
    {
        if ( $jaahytiedot[$seka][$x][$syy] == $sqlrivi[Koodi] ) $jaahytiedot[$seka][$x][$syy] =
            utf8_decode($sqlrivi[Selitys]);
    }
    $seka=$temp;
}
$seka=$temp;
$apu=0;
}
if (count($matsit)>3)
{
    $URLI=$PHP_SELF."?".$_SERVER['QUERY_STRING'];
    $sivu = explode("&viimininen",$URLI);
    $rivit=1;
    for($x=0;$x<count($matsit);$x++)
    {
        if (($x%4) == 0)$rivit++;
    }
    echo "<TABLE align=\"center\" border=\"0\" cellpadding=\"0\" cellspacing=\"0\" width=\"800\"><TR><TD>";
    for($x=1;$x<$rivit;$x++)
    {
        if (($x*4) > count($matsit))
        {
            $Loppu= count($matsit);
            $Alku=(x-1)*4;
        }
        else
        {

```


Liite 1 20(26)

```

        $Loppu=( $\$x*4$ );
        $Alku=$Loppu-4;
    }
    $Kuvaus="";
    for( $\$z=\$Alku;\$z<\$Loppu;\$z++$ )
    {
        $Kuvaus .= $matsit[$z][Ottelu]." ".$matsit[$z][Lopputulos];
        if ( $\$z < (\$Loppu-1)$ ) $Kuvaus .= ", ";
    }
    $apu=1;
    echo "<a style='color: #FF0000' href='\$sivu[0]&Viiminen=$Loppu&eka=$Alku&apu=$apu'
    title='\$Kuvaus' >".( $\$x*4-3$ )."-";
    if (( $\$x*4$ ) > count($matsit)) echo count($matsit)."</a>." ";
    else echo ( $\$x*4$ )."</a>." ";
    echo " ";
}
echo "</TD></TR><TABLE>";
}
else
{
    $seka=0;
    $Viiminen= count($matsit);
}

echo "<TABLE align='center' border='0' cellpadding='0' cellspacing='0' width='800'><TR><TD>";
$temp=$seka;
for($seka;$seka<$Viiminen;$seka++)
{
    echo "<a style='color: #FF0000' href='\"#\$seka\" >".$matsit[$seka][Ottelu]." ".$matsit[$seka][Lopputulos]."</a>";
    echo " ";
}
echo "</TD></TR><TABLE>";
$seka=$temp;

for($seka;$seka<$Viiminen;$seka++)
{
    $Koti=0;
    $vieras=0;
    $jaahyteraja=0;
    $jaahyterat[0]=0;
    $jaahyterat[1]=0;
    $jaahyterat[2]=0;
    $maalit_JA=0;
    $maalit_RL=0;

    $maalit_erät[0] = $matsit[$seka][Kotiera1] + $matsit[$seka][Vierasera1];
    $maalit_erät[1] = $matsit[$seka][Kotiera2] + $matsit[$seka][Vierasera2];
    $maalit_erät[2] = $matsit[$seka][Kotiera3] + $matsit[$seka][Vierasera3];
    $maalit_JA = $matsit[$seka][KotiJA] + $matsit[$seka][VierasJA];
    $maalit_RL = $matsit[$seka][KotiRL] + $matsit[$seka][VierasRL];

    for( $\$z=0;\$z<count(\$Jaahytiedot[\$seka]);\$z++$ )
    {
        if ( $\$Jaahytiedot[\$seka][\$z][Era] == 1$ ) $jaahyterat[0]++;
        if ( $\$Jaahytiedot[\$seka][\$z][Era] == 2$ ) $jaahyterat[1]++;
        if ( $\$Jaahytiedot[\$seka][\$z][Era] == 3$ ) $jaahyterat[2]++;
        if ( $\$Jaahytiedot[\$seka][\$z][Era] == "JA"$ ) $jaahyteraja++;
    }

    for( $\$y=0;\$y<count(\$Maalitiedot[\$seka]);\$y++$ )
    if ( $\$Maalitiedot[\$seka][\$y][Koodi] == "-"$ ) $Maalitiedot[$seka][\$y][Koodi] = "";

    ?>
    <Script Language="JavaScript">
    kp[<?php echo $seka; ?>]= "pelaajat.php?joukkue=<?php echo
    $matsit[$seka][Kotijoukkue]." &id=".$matsit[$seka][PeliID];?>";
    vp[<?php echo $seka; ?>]= "pelaajat.php?joukkue=<?php echo
    $matsit[$seka][Vierasjoukkue]." &id=".$matsit[$seka][PeliID];?>";
    </Script>
    <?php /*RULES=NONE FRAME=BOX*/
    echo "<TABLE border='1' RULES=NONE FRAME=BOX class='otttul' align='center' cellpadding='1'
    cellspacing='0' width='800'>";
    echo "<TR><TD colspan='2'><FONT size='5'><B1><A NAME =\"$seka\">Ottelu: </A></B1>";
    echo "<a style='color: #FF0000' href='\"javascript:load(kp[\$seka])\">".$matsit[$seka][Kotijoukkue];
    echo "</a> - <a style='color: #FF0000' href='\"javascript:load(vp[\$seka])\">".$matsit[$seka][Vierasjoukkue];

```

Liite 1 21(26)

```

echo "</a> ".$matsit[$seka][Lopputus]."</FONT></TD>";
echo "<TD colspan=\2\" ><FONT size=\5\"><B1>Voittaja:
</B1>".$matsit[$seka][Voittaja]."</FONT></TD></TR>\n";
echo "<TR><TD><B>Päivämäärä: </B>".$matsit[$seka][Paivamaara]."</TD><TD ><B>Pelipaikka:
</B>".$matsit[$seka][Pelipaikka]."</TD>";
echo "<TD><B>Alkamisaika: </B>".$matsit[$seka][Alkamisaika]."</TD><TD><B>Loppumisaika:
/B>".$matsit[$seka][Loppumisaika]."</TD></TR>\n";
echo "<TR><TD colspan=\2\"><B>Kausi: </B>".$matsit[$seka][Kausi]."</TD>";
echo "<TD colspan=\2\"><B>Tuomarit: </B>".$matsit[$seka][Tuomari1];
if ($matsit[$seka][Tuomari1] != "") echo " ja ";
echo $matsit[$seka][Tuomari2]."</TD></TR>\n";
echo "<TR><TD colspan=\2\"><B>".$matsit[$seka][Kotijoukkue]." aikalisä:
</B>".$matsit[$seka][Kotiaikalisa]."</TD>";
echo "<TD colspan=\2\"><B>".$matsit[$seka][Vierasjoukkue]." aikalisä:
</B>".$matsit[$seka][Vierasikalisa]."</TD></TR>\n";
for($x=0;$x<$matsit[$seka][EraMaara];$x++)
{
    $ktempera = "Kotiera".($x+1);
    $vtempera = "Vierasera".($x+1);
    echo "<TR><TD colspan=\4\"><FONT size=\4\"><B>".($x+1).". erä:
    </B>".$matsit[$seka][$ktempera]."- ".$matsit[$seka][$vtempera]."</FONT></TD></TR>\n";
    if ($maalit_erät[$x] == 0) echo "<TR><TD colspan=\4\"><B>Maaliton</B></TD></TR>";
    else
    {
        echo "<TR><TD colspan=\4\"><B>Maalitiedot:</B></TD></TR>\n";
        for($z=0;$z<count($Maalitiedot[$seka]);$z++)
        {
            if ($Maalitiedot[$seka][$z][Era] == ($x+1))
            {
                echo "<TR><TD >&nbsp;<B>".$Maalitiedot[$seka][$z][Aika]."</B>
                B2>".$Maalitiedot[$seka][$z][Joukkue];
                if ($Maalitiedot[$seka][$z][Numero] != "ERL") echo "
                ".$Maalitiedot[$seka][$z][Koodi]."</B2>";
                else echo " ERL</B2>";
                echo "<TD><TD colspan=\3\"> ";
                if ($Maalitiedot[$seka][$z][Numero] != "ERL")if
                ($Maalitiedot[$seka][$z][Joukkue]==$matsit[$seka][Kotijoukkue]) $Koti++;
                if ($Maalitiedot[$seka][$z][Numero] != "ERL")if
                ($Maalitiedot[$seka][$z][Joukkue]==$matsit[$seka][Vierasjoukkue]) $vieras++;
                if ($Maalitiedot[$seka][$z][Numero] != "ERL") echo "<B>$Koti</B>-
                <B>$vieras</B>";
                echo "<A style=\color: #FF0000\"
                href=\?pknimi=\".$Maalitiedot[$seka][$z][Tekija]."&pkjoukkue=\".$Maalitiedot
                [$seka][$z][Joukkue].">".$Maalitiedot[$seka][$z][Tekija]."<A> <B></B>";
                if ($Maalitiedot[$seka][$z][Syottaja] != "-") echo "<A style=\color: #FF0000\"
                href=\?pknimi=\".$Maalitiedot[$seka][$z][Syottaja]."&pkjoukkue=\".$Maalitiedot
                [$seka][$z][Joukkue].">".$Maalitiedot[$seka][$z][Syottaja]."<A><B></B>
                </TD><TD></TD></TR>\n";
                else echo
                $Maalitiedot[$seka][$z][Syottaja]."<B></B></TD><TD></TD></TR>\n";
            }
        }
    }
}
echo "<TR><TD colspan=\4\"><BR></TD></TR>";
if ($jaahyerat[$x] > 0)
{
    echo "<TR><TD colspan=\4\"><FONT
    size=\3\"><B>Rangaistukset:</B></FONT></TD></TR>\n";
    for($y=0;$y<count($Jaahytiedot[$seka]);$y++)
    {
        if ($Jaahytiedot[$seka][$y][Era] == ($x+1))
        {
            echo "<TR><TD>&nbsp;<B>".$Jaahytiedot[$seka][$y][Alkuaika]."</B>
            <B2>".$Jaahytiedot[$seka][$y][Joukkue]."</B2></TD>";
            echo "<TD colspan=\3\"><A style=\color: #FF0000\"
            href=\?pknimi=\".$Jaahytiedot[$seka][$y][Numero]."&pkjoukkue=\".$Jaahytiedot
            [$seka][$y][Joukkue].">".$Jaahytiedot[$seka][$y][Numero]."<A>,
            ".$Jaahytiedot[$seka][$y][Kesto]."<B> min.
            </B>".$Jaahytiedot[$seka][$y][Syy]."</TD></TR>";
        }
    }
}
echo "<TR><TD colspan=\4\"><BR></TD></TR>";
}
}
if ($matsit[$seka][KotiJA] != "-" || $matsit[$seka][VierasJA] != "-")
{

```


Liite 1 23(26)

```

else echo $Maalitiedot[$eka][$z][$syottaja]."<B></B></TD><TD></TD></TR>\n";
    }
}
}
/*
print "<TR><TD colspan=\\"5\"><BR></TD></TR><TR><TD colspan=\\"5\"><FONT
size=\\"3\"><B>Rangaistukset:</B></FONT></TD></TR>\n";
for($z=0;$z<Count($Jaahytiedot[$eka]);$z++)
{
    print "<TR><TD colspan=\\"2\">".$Jaahytiedot[$eka][$z][$joukkue].",
    ".$Jaahytiedot[$eka][$z][$numero].", ".$Jaahytiedot[$eka][$z][$syy]."</TD><TD><B>Pituus(min):
</B>".$Jaahytiedot[$eka][$z][$kesto]."</TD>";
    print "<TD><B>Alkamisaika: </B>".$Jaahytiedot[$eka][$z][$alkuaika]."</TD><TD><B>Loppumisaika:
</B>".$Jaahytiedot[$eka][$z][$loppuaika]."</TD></TR>\n";
}
*/
?>
<BR></TABLE><TABLE align="center" border="0" cellpadding="0" cellspacing="0" width="800">
<TR><TD ><A style="color: #FF0000" href="#Alku">Ylös</A></TD></TR></TABLE>
<?php
}
}
else
{
    ?>
<TABLE align="center" border="0" cellpadding="0" cellspacing="0" width="800">
<TR><TD class="otteluvirhe">Pelej&#228; ei l&#246;ytynty valituilla hakuehdoilla!</TD></TR></TABLE>
<?php
}
}
if ($Tilasto_valinta == "Sarjataulukot")
{
    $kysely .="";
    $sarjataulukysely = "SELECT * FROM ottelut ";
    $sarjataulukysely .= $kysely;
    $stamppi=0;
    // $sote = mysql_connect( $localhost , $kayttaja );
    //if ( ! $sote ) die( "Tietokantaan ei saatu yhteyttä!" );
    mysql_select_db( $tietokanta, $sote ) or die ( "Tietokantaa $tietokanta: "." ei saatu valittua".mysql_error() );
    $sqlhakutieto = mysql_query( $sarjataulukysely, $sote );
    while ( $sqlrivi = mysql_fetch_array( $sqlhakutieto ) )
    {
        $sarjautietoottelut[ ][ID] = $sqlrivi[PeliID];
    }
    if (count($sarjautietoottelut) != 0)
    {
        $sqlhakutieto = mysql_query( $sarjataulukysely, $sote );
        while ( $sqlrivi = mysql_fetch_array( $sqlhakutieto ) )
        {
            for($x=0;$x<count($KaikkiJoukkueet);$x++)
            {
                if ($KaikkiJoukkueet[$x][Joukkue]==$sqlrivi[Kotijoukkue] )
                {
                    $stamppi=1;
                }
            }
            if ($stamppi==0)
            {
                $KaikkiJoukkueet[ ][Joukkue] = utf8_decode($sqlrivi[Kotijoukkue]);
            }
            $stamppi=0;
            for($x=0;$x<count($KaikkiJoukkueet);$x++)
            {
                if ($KaikkiJoukkueet[$x][Joukkue]==$sqlrivi[Vierasjoukkue] )
                {
                    $stamppi=1;
                }
            }
            if ($stamppi==0)
            {
                $KaikkiJoukkueet[ ][Joukkue] = utf8_decode($sqlrivi[Vierasjoukkue]);
            }
            $stamppi=0;
        }
        for( $x=0;$x<count($KaikkiJoukkueet);$x++)
        {

```

Liite 1 24(26)

```

$KaikkiJoukkueet[$x][Pisteet] =0;
$KaikkiJoukkueet[$x][Ottelut] =0;
$KaikkiJoukkueet[$x][Voitot] =0;
$KaikkiJoukkueet[$x][Tasapelit] =0;
$KaikkiJoukkueet[$x][Tappiot] =0;
$KaikkiJoukkueet[$x][Tehdyt] =0;
$KaikkiJoukkueet[$x][Paastetyt] =0;
$KaikkiJoukkueet[$x][Maaliero] =0;
$KaikkiJoukkueet[$x][Pisteet] =0;
}
$Sqlhakutieto = mysql_query( $sarjataulukysely, $osote );
while ($sqlrivi = mysql_fetch_array( $sqlhakutieto ))
{
    for( $x=0;$x<count($KaikkiJoukkueet);$x++)
    {
        if ($sqlrivi[Voittaja] == "Tasapeli" && $KaikkiJoukkueet[$x][Joukkue] ==
        $sqlrivi[Kotijoukkue])
        {
            $KaikkiJoukkueet[$x][Pisteet] +=1;
            $KaikkiJoukkueet[$x][Tasapelit] +=1;
        }
        if ($sqlrivi[Voittaja] == "Tasapeli" && $KaikkiJoukkueet[$x][Joukkue] ==
        $sqlrivi[Vierasjoukkue])
        {
            $KaikkiJoukkueet[$x][Pisteet] +=1;
            $KaikkiJoukkueet[$x][Tasapelit] +=1;
        }
        if ($KaikkiJoukkueet[$x][Joukkue] == $sqlrivi[Voittaja] && $KaikkiJoukkueet[$x][Joukkue]
        == $sqlrivi[Kotijoukkue])
        {
            for( $z=0;$z<count($KaikkiJoukkueet);$z++)
            if( $KaikkiJoukkueet[$z][Joukkue] == $sqlrivi[Vierasjoukkue])
            $KaikkiJoukkueet[$z][Tappiot] +=1;
            $KaikkiJoukkueet[$x][Pisteet] +=2;
            $KaikkiJoukkueet[$x][Voitot] +=1;
        }
        if ($KaikkiJoukkueet[$x][Joukkue] == $sqlrivi[Voittaja] && $KaikkiJoukkueet[$x][Joukkue]
        == $sqlrivi[Vierasjoukkue])
        {
            for( $z=0;$z<count($KaikkiJoukkueet);$z++)
            if($KaikkiJoukkueet[$z][Joukkue] == $sqlrivi[Kotijoukkue])
            $KaikkiJoukkueet[$z][Tappiot] +=1;
            $KaikkiJoukkueet[$x][Pisteet] +=2;
            $KaikkiJoukkueet[$x][Voitot] +=1;
        }
    }

    //Maalien laitto joukkueelle
    if ($KaikkiJoukkueet[$x][Joukkue] == $sqlrivi[Kotijoukkue])
    {
        $KaikkiJoukkueet[$x][Tehdyt] +=
        ($sqlrivi[Kotiera1]+$sqlrivi[Kotiera2]+$sqlrivi[Kotiera3]+$sqlrivi[KotiJA]);
        $KaikkiJoukkueet[$x][Paastetyt] +=
        ($sqlrivi[Vierasera1]+$sqlrivi[Vierasera2]+$sqlrivi[Vierasera3]+$sqlrivi[VierasJA]);
        $KaikkiJoukkueet[$x][Ottelut] += 1;
    }
    if ($KaikkiJoukkueet[$x][Joukkue] == $sqlrivi[Vierasjoukkue])
    {
        $KaikkiJoukkueet[$x][Tehdyt] +=
        ($sqlrivi[Vierasera1]+$sqlrivi[Vierasera2]+$sqlrivi[Vierasera3]+$sqlrivi[VierasJA]);
        $KaikkiJoukkueet[$x][Paastetyt] +=
        ($sqlrivi[Kotiera1]+$sqlrivi[Kotiera2]+$sqlrivi[Kotiera3]+$sqlrivi[KotiJA]);
        $KaikkiJoukkueet[$x][Ottelut] += 1;
    }
}
for( $x=0;$x<count($KaikkiJoukkueet);$x++)
$KaikkiJoukkueet[$x][Maaliero]=$KaikkiJoukkueet[$x][Tehdyt]-$KaikkiJoukkueet[$x][Paastetyt];
Jarjestely(&$KaikkiJoukkueet,"Pisteet","Maaliero","Tehdyt");
?>
<BR>
<TABLE align="center" class="tulosost" border="1" cellpadding="1" cellspacing="1" width="800">
<TR><TD class="tulosostsikko">Nro:</TD><TD class="tulosostsikko">Joukkue:</TD><TD
class="tulosostsikko">Ottelut:</TD>
<TD class="tulosostsikko">Voitot:</TD><TD class="tulosostsikko">Tasapelit:</TD><TD
class="tulosostsikko">Tappiot:</TD>
<TD class="tulosostsikko">Tehd.Maal.:</TD><TD class="tulosostsikko">Pääst.Maal.:</TD><TD
class="tulosostsikko">Maaliero:</TD>

```

Liite 1 25(26)

```

<TD class="tulosotsikko">Pisteet:</TD></TR>
<?php
$z=1;
for( $x=0;$x<count($KaikkiJoukkueet);$x++)
{
    echo "<TR><TD>". $z++. "</TD>";
    echo "<TD>". $KaikkiJoukkueet[$x][Joukkue]. "</TD>";
    echo "<TD>". $KaikkiJoukkueet[$x][Ottelut]. "</TD>";
    echo "<TD>". $KaikkiJoukkueet[$x][Voitot]. "</TD>";
    echo "<TD>". $KaikkiJoukkueet[$x][Tasapelit]. "</TD>";
    echo "<TD>". $KaikkiJoukkueet[$x][Tappiot]. "</TD>";
    echo "<TD>". $KaikkiJoukkueet[$x][Tehdyt]. "</TD>";
    echo "<TD>". $KaikkiJoukkueet[$x][Paastetyt]. "</TD>";
    echo "<TD>". $KaikkiJoukkueet[$x][Maaliero]. "</TD>";
    echo "<TD>". $KaikkiJoukkueet[$x][Pisteet]. "</TD></TR>";
}
?>
</TABLE><TABLE align="center" border="0" cellpadding="0" cellspacing="0" width="800">
<TR><TD><A style="color: #FF0000" href="#Alku">Ylös</A></TD></TR></TABLE>
<?php
}
else
{
    ?>
<TABLE align="center" border="0" cellpadding="0" cellspacing="0" width="800">
<TR><TD class="otteluvirhe">Pelejä ei löytynyt valituilla hakuehdoilla!</TD></TR></TABLE>
<?php
}
}
//echo $pknimi;

if ($pknimi != "" && $pkjoukkue != "" )
{
    //Spkkaudet= array();

    $kysely_nimi = utf8_encode($pknimi);
    $kysely_joukkue = utf8_encode($pkjoukkue);
    $pelaajakorttikysely = "SELECT * FROM pelaajatiedot WHERE Nimi='$kysely_nimi' AND
    Joukkue='$kysely_joukkue'";
    //echo $pelaajakorttikysely. "<br>";
    // $osote = mysql_connect( $host , $kayttaja );
    //if ( ! $osote ) die( "Tietokantaan ei saatu yhteyttä!" );
    mysql_select_db( $tietokanta, $osote ) or die ( "Tietokantaa $tietokanta: ". " ei saatu valittua".mysql_error() );
    $sqlhakutieto = mysql_query( $pelaajakorttikysely, $osote );
    while ( $sqlrivi = mysql_fetch_array( $sqlhakutieto ) )
    {
        Taulukointi(&$pkkaudet,$sqlrivi,"Kausi");
    }
    sort($pkkaudet);
    $sqlhakutieto = mysql_query( $pelaajakorttikysely, $osote );
    while ( $sqlrivi = mysql_fetch_array( $sqlhakutieto ) )
    {
        for($x=0;$x<count($pkkaudet);$x++)
        {
            if ( $sqlrivi[Kausi] == $pkkaudet[$x] )
            {
                $spkporssi_pelaaja[$x][Maalit] += $sqlrivi[Maalit];
                $spkporssi_pelaaja[$x][Syotot] += $sqlrivi[Syotot];
                $spkporssi_pelaaja[$x][Pisteet] += $sqlrivi[Pisteet];
                $spkporssi_pelaaja[$x][Jaahyt] += $sqlrivi[Jaahyt];
                $spkporssi_pelaaja[$x][Ottelut] += 1;
                $spkporssi_pelaaja[$x][YVMaalit] += $sqlrivi[YVMaalit];
                $spkporssi_pelaaja[$x][YVSyotot] += $sqlrivi[YVSyotot];
                $spkporssi_pelaaja[$x][YVPisteet] += $sqlrivi[YVPisteet];
                $spkporssi_pelaaja[$x][AVMaalit] += $sqlrivi[AVMaalit];
                $spkporssi_pelaaja[$x][AVSyotot] += $sqlrivi[AVSyotot];
                $spkporssi_pelaaja[$x][AVPisteet] += $sqlrivi[AVPisteet];
                $spkporssi_pelaaja[$x][RLYritykset] += $sqlrivi[RLYritykset];
                $spkporssi_pelaaja[$x][RLMaalit] += $sqlrivi[RLMaalit];
                $spkporssi_pelaaja[$x][Joukkue] = $sqlrivi[Joukkue];
            }
        }
        $spkporssi_pelaaja_yht[Ottelut] += 1;
        $spkporssi_pelaaja_yht[Maalit] += $sqlrivi[Maalit];
        $spkporssi_pelaaja_yht[Syotot] += $sqlrivi[Syotot];
        $spkporssi_pelaaja_yht[Pisteet] += $sqlrivi[Pisteet];
    }
}
}

```

Liite 1 26(26)

```

        $pkporssi_pelaaja_yht[Jaahyt] += $sqlrivi[Jaahyt];
    }

?>
<TABLE class="tulostus" align="center" border="1" cellpadding="0" cellspacing="1" width="800">

<TR><TD colspan="8" class="tulotsikko">Pelaajakortti</TD></TR>
<TD colspan="8" ><?php echo $pknimi.", ".$pkjoukkue; ?></TD></TR>
<TR><TD class="tulotsikko">Kaudet:</TD><TD class="tulotsikko">Ottelut:</TD><TD
class="tulotsikko">Maalit:</TD>
<TD class="tulotsikko">Syötöt:</TD><TD class="tulotsikko">Pisteet:</TD><TD
class="tulotsikko">Pist./Ott.:</TD><TD class="tulotsikko">Jäähyt:</TD><TD class="tulotsikko">Min./Ott.:</TD>
<?php
for($x=0;$x<count($pkkaudet);$x++)
{
    echo "<TR><TD>$pkkaudet[$x]</TD>";
    echo "<TD>".$pkporssi_pelaaja[$x][Ottelut]."</TD>";
    echo "<TD>".$pkporssi_pelaaja[$x][Maalit]."</TD>";
    echo "<TD>".$pkporssi_pelaaja[$x][Syotot]."</TD>";
    echo "<TD>".$pkporssi_pelaaja[$x][Pisteet]."</TD><TD>";
    printf ( "% .2f", $pkporssi_pelaaja[$x][Pisteet]/$pkporssi_pelaaja[$x][Ottelut]);
    echo "<TD><TD>".$pkporssi_pelaaja[$x][Jaahyt]."</TD><TD>";
    printf ( "% .2f", $pkporssi_pelaaja[$x][Jaahyt]/$pkporssi_pelaaja[$x][Ottelut]);
    echo "<TD></TR>";
}
echo "<TR><TD class='tulotsikko'>Yhteensä:</TD><TD class='tulotsikko' colspan='7'><BR></TD></TR>";
echo "<TR><TD>Kaikki kaudet</TD><TD>$pkporssi_pelaaja_yht[Ottelut]";
echo "<TD>$pkporssi_pelaaja_yht[Maalit]";
echo "<TD>$pkporssi_pelaaja_yht[Syotot]";
echo "<TD>$pkporssi_pelaaja_yht[Pisteet]";
echo "<TD>";
printf ( "% .2f", $pkporssi_pelaaja_yht[Pisteet]/$pkporssi_pelaaja_yht[Ottelut]);
echo "<TD>$pkporssi_pelaaja_yht[Jaahyt]";
echo "<TD>";
printf ( "% .2f", $pkporssi_pelaaja_yht[Jaahyt]/$pkporssi_pelaaja_yht[Ottelut]);
?>
</TABLE>
<TABLE align="center" border="0" cellpadding="0" cellspacing="1" width="800">
<tr><TD align="center">
<a style="color: #FF0000" href="javascript:history.back();">Takaisin</a>
</TD></tr>
</TABLE>

<?php

}

mysql_close( $osote );
?>
</body>

</html>

```