

# Cbook : Asiakastietokanta CRUD-toiminnallisuudella

Tässä kappaleessa esitetään pienen Cbook-sovelluksen (CustomerBook) perustoiminnot asiakastietokannan sisällön listaamiseksi ja tietuiden lisäämiseksi. Esimerkkietokantana käytetään aiemmin esitettyä asiakastietokantaa ja sen `customer`-taulua. Harjoitustehtäväksi jätetään täydennettäväksi sovellukseen tässä toteuttamatta jääneet osat CRUD-toiminnallisuudesta (Update ja Delete) sekä asiakkaan tekstihaku nimen perusteella.

Esitettävien ohjelmakoodien tarkempi `toiminnallisuus` selittää puuttuvilta osin lähiopetuksessa ja videoluennoilla. Tässä esitetään perusrakenne ja mahdollisuus kopioida ohjelmakoodit kokeiltavaksi ja testattavaksi. On erittäin tärkeää, että **itse käytännössä** kokeilet ja muuntelet annettuja esimerkkejä hallitaksesi sisällön.

Ohjelmaesimerkit saat toimimaan, kun käytät **oman tietokannan** `customer`-taulua ja ohjelmaesimerkkien tietokantayhteystietoina **omia** tietojaasi. Ohjelmaesimerkkien esimerkkikäyttäjätunnus `N1234` tulee vaihtaa omaan tunnukseesi.

Ohjelman päänäköymä on toteutettu tiedostoon `cbook-list.php` ja se näyttää tältä

Cbook by QaD Solutions

[\[ Listaa \]](#) [\[ Lisää \]](#)

| Id | Name         | Birth_date | Created_at          |
|----|--------------|------------|---------------------|
| 1  | Guru Ken     | 1955-05-05 | 22.09.2019 15:31:36 |
| 2  | Alainen Kim  | 1966-06-06 | 22.09.2019 15:31:36 |
| 3  | Ainen Sani   | 1977-07-07 | 22.09.2019 15:31:36 |
| 4  | Vainio Vilja | 1988-08-08 | 22.09.2019 15:31:36 |
| 5  | Tana Ruu     | 1999-09-09 | 22.09.2019 15:31:36 |

Ohjelman lähdekoodi on esitetty ohessa.

- Näkymän layout on toteutettu riveillä 1-7
- Sivun sisältö haetaan tietokannasta `page_content()` -funktiolla `$customerObj`

- Olio `$customerObj` liittyvä toiminnallisuus on ohjelmoitu tiedostoon `/home/N1234/db-config/CustomerDb.php` (rivi 15) Huomaa, että ko. tiedosto ei ole tallennettu `public_html`-kansioon, koska siihen on tallennettu myös tietokantayhteyteen liittyviä autentikointitietoja.
- Funktiolla `do_html_table()` muotoillaan `$customers`-taulukossa objekteina tallennetut asiakastietueet HTML-sivulle.
- `getCustomers()` -metodille välitettävä `$tyhja_hakusana` on tulevia tarpeita varten. Sitä ei käytetä tässä.
- Muut viitatu tiedostot esitetään jäljempänä

## cbook-list.php

```

1  <title>cbook-list.php</title>
2  <link rel="stylesheet" href="tyyli.css" type="text/css">
3
4  <div id='container'>
5      <?php include('navbar.php');?>
6      <?php page_content();?>
7  </div>
8
9  <?php
10 /*****          PHP-toiminnot          *****/
11
12 /*****   Datan hakeminen tietokannasta   *****/
13 function page_content() {
14
15     require_once("/home/N1234/db-config/CustomerDb.php");
16     $customerObj = new CustomersDb();
17     $tyhja_hakusana = '';
18     $customers = $customerObj->getCustomers($tyhja_hakusana);
19
20     if (count($customers)>=1) echo do_html_table($customers);
21     //var_dump($customers); // ks. tarvittaessa
22 }
23
24 /*****   Asiakaslista HTML-taulukkona   *****/
25 function do_html_table($customers) {
26     $html = "<table>";
27
28     // Otsikkorivi
29     $html .= "<tr>";
30     foreach ($customers[0] as $key => $value) {
31         $html .= "<th>" . ucfirst($key) . "</th>";
32     }
33     $html .= "</tr>";

```

```

33
34     // Tulostusrivit
35     foreach ($customers as $customer) {
36         $html .= "<tr>" .
37             "<td>$customer->id</td>" .
38             "<td>$customer->name</td>" .
39             "<td>$customer->birth_date</td>" .
40             "<td>$customer->created_at</td>" .
41             "</tr>";
42     }
43     $html .= "</table>";
44
45     return $html;
46 }
47

```

## Luokka tietokannan käsittelyyn: CustomerDb.php

- Tietokannan käsittelyä varten luodaan erillinen luokka `CustomersDb`, joka sisältää yhteydenmuodostuksen tietokantaan sekä tarvittavat metodit tietokannan sisällön listaamiseksi ja uuden tietueen lisäämiseksi
- Huomaa, että ko. tiedosto ei ole tallennettu `public_html`-kansioon, koska siihen on tallennettu myös tietokantayhteyteen liittyviä autentikointitietoja.
- Metodeissa on käytetty jo aiemmin esiteltäviä menetelmiä tietokannan sisällön hallintaan
- Huomaa, että `getCustomers()` -metodissa tietueet haetaan objekteina (rivi 38)
- `getCustomers()` -metodille välitettävä `$searchterm` on tulevia tarpeita varten. Sen ollessa tyhjänä haetuksi tulee kaikki asiakkaat (tietueet).

/home/N1234/db-config/CustomerDb.php

```

1  <?php
2
3  class CustomersDb {
4      private $dbConnection;
5
6      public function __construct() {
7          try {
8              $this->dbConnection = new PDO('mysql:host=mysql.labra
9                  'N1234', 'se-pitka-mysql-salasana');
10             //$this->dbConnection->setAttribute(PDO::ATTR_ERRMODE
11             //$this->dbConnection->setAttribute(PDO::ATTR_ERRMODE
12             $this->dbConnection->setAttribute(PDO::ATTR_ERRMODE,

```

```

12     } catch(PDOException $ex) {
13         echo "ErrMsg to enduser!<hr>";
14         echo "CatchErrMsg: " . $ex->getMessage() . "<hr>";
15         //logError($ex->getMessage());
16     }
17 }
18
19
20 /***** getCustomers
21 public function getCustomers($searchterm) {
22     $customers = array();
23     $sql = <<<EndOfSQL
24         SELECT
25             id,
26             name,
27             birth_date,
28             DATE_FORMAT(created_at, "%d.%m.%Y %H:%i:%s") AS created_at
29         FROM customer WHERE name
30         LIKE :searchterm
31     EndOfSQL;
32
33     $resultObj = $this->dbConnection->prepare($sql);
34     $resultObj->bindValue(':searchterm', "%$searchterm%", PDO::PARAM_STR);
35     $resultObj->execute();
36
37     while ($obj = $resultObj->fetch(PDO::FETCH_OBJ)) {
38         $customers[] = $obj;
39     }
40
41     return $customers;
42 }
43
44
45 /***** addCustomer
46
47 public function addCustomer($name, $birth_date) {
48     $addResult = 0;
49
50
51     $sql = <<<EndOfSQL
52         INSERT INTO customer (name, birth_date, created_at) VALUES
53         (:name, :birth_date, (select now()))
54     EndOfSQL;

```

```

55
56
57     $result = $this->dbConnection->prepare($sql);
58     $result->bindValue(':name', $name, PDO::PARAM_STR);
59     $result->bindValue(':birth_date', $birth_date, PDO::PARAM
60     $result->execute();
61
62     if ($result->rowCount() == 1) {
63         $addResult = $this->dbConnection->lastInsertId();
64     } else {
65         $addResult = 0; // voi katsoa myös echo $this->dbConnec
66     }
67
68     return $addResult;
69 }
70
71 }
72

```

Sovelluksen navigointivalikko on seuraava

### navbar.php

```

1 <div style="text-align: right; margin-top: 10px;padding-bottom
2 <span style="background-color: #ddd;">Cbook</span>
3 <em>by QaD Solutions</em>
4 </div>
5     [ <a href='cbook-list.php'>Listaa</a> ]
6     [ <a href='cbook-addform.php'>Lisää</a> ]
7
8 <hr>

```

Sovelluksen tyylitiedosto on seuraava

### tyyli.php

```

1 body {font: 100% arial, helvetica, sans-serif;}
2
3 h2, h3 {background-color:#e7ecf1; text-align: center}
4
5 table {margin: auto;}
6 #container {width: 400px; margin:auto;}
7

```

```

8   p {padding: 0.1em; margin: 0.1em;}
9
10  .boxi {
11    background-color: #f1f1f1; border: 1px solid #e5e5e5;
12    padding: 10px 0px 10px 25px;
13  }
14
15  tr:nth-child(odd) {background: #ccc}
16  tr:nth-child(even) {background: #fff}
17  tr:nth-child(1) {background: #e7ecf1}

```

## Lisäyslomake

Lisäyslomake näyttää käyttöliittymältään seuraavalta:

*Cbook by QaD Solutions*

[\[ Listaa \]](#) [\[ Lisää \]](#)

---

**Lisää asiakas**

**Nimi**

**Syntymäpäivä**

onnistuneen lisäyksen jälkeen ladataan automaattisesti listausnäkyvä, jossa lisätyt asiakastiedot näkyvät

[\[ Listaa \]](#) [\[ Lisää \]](#)

## Listaa asiakkaat

| Name                         | Birth_date | Created_at          |
|------------------------------|------------|---------------------|
| <a href="#">Guru Ken</a>     | 1955-05-05 | 22.09.2019 15:31:36 |
| <a href="#">Alainen Kim</a>  | 1966-06-06 | 22.09.2019 15:31:36 |
| <a href="#">Ainen Sani</a>   | 1977-07-07 | 22.09.2019 15:31:36 |
| <a href="#">Vainio Vilja</a> | 1988-08-08 | 22.09.2019 15:31:36 |
| <a href="#">Tana Ruu</a>     | 1999-09-09 | 22.09.2019 15:31:36 |
| <a href="#">Rahainen Muu</a> | 2011-11-11 | 23.09.2019 21:38:29 |

Lisäyslomakkeen PHP-toteutus on suoraviivainen. Testauksen helpottamiseksi syntymäpäivä-kenttään on asetettu sopiva oletusarvo, jota voit halutessasi muuttaa:

### cbook-addform.php

```

1  <title>cbook-addform.php</title>
2  <link rel="stylesheet" href="tyyli.css" type="text/css">
3
4  <div id='container'>
5      <?php include('navbar.php');?>
6      <?php echo blank_form();?>
7  </div>
8
9
10 <?php
11 /***** PHP-toiminnot *****/
12
13 /***** Tyhjä lomake *****/
14 function blank_form() {
15
16     $form = <<<EndOfForm
17     <h3>Lisää asiakas</h3>
18
19     <div class="boxi">
20     <form action="cbook-save.php" method="post">
21         Nimi<br>
22         <input type="text" name="name" placeholder="Sukunimi Et

```

```

23         Syntymäpäivä<br>
24         <input type="text" name="birth_date" value="2011-11-11"
25         <input type="submit" name="nappi" value="Tallenna">
26     </form>
27 </div>
28 EndOfForm;
29
30     return $form;
31 }

```

## Asiakastietueen tallentaminen

Kun tyhjä lomake on täytetty ja klikattu **Tallenna**-painiketta, lomakkeen tiedot tallennetaan tietokantaan `cbook-save.php`-tiedostoon kirjoitetun ohjelmakoodin avulla. Jos tallentaminen ei onnistu, tulostetaan näkyville `$errMsg`-muuttujan arvo. Varsinainen tallentaminen eli lisääminen tietokantaan tapahtuu aiemmin esitettyyn tiedostoon `/home/ara/db-config/CustomerDb.php` kirjoitetulla `addCustomer()`-metodilla.

### cbook-save.php

```

1  <?php $errMsg = save_form(); ?>
2
3
4  <title>cbook-save.php</title>
5  <link rel="stylesheet" href="tyyli.css" type="text/css">
6
7  <div id='container'>
8      <?php include('navbar.php');?>
9      <?php echo $errMsg;?>
10 </div>
11
12
13 <?php
14 /***** PHP-toiminnot *****/
15
16 /***** Tallennetaan tiedot *****/
17 function save_form() {
18
19     require_once("/home/ara/db-config/CustomerDb.php");
20     $customerObj = new CustomersDb();

```



```
21
22     $errMsg = '';
23     $success = 0;
24
25     // Datat vastaanotto
26     $name      = isset($_POST['name'])      ? $_POST['name']
27     $birth_date = isset($_POST['birth_date']) ? $_POST['birth_date']
28
29     // Alkeellista tarkistusta lisäystä varten
30     if (strlen($name)>=1 AND strlen($birth_date)>=1) {
31         $success = $customerObj->addCustomer($name, $birth_date);
32     }
33
34     if ($success) {
35         header("Location: http://" . $_SERVER['HTTP_HOST']
36             . dirname($_SERVER['PHP_SELF']) . '/'
37             . "cbook-list.php");
38     } else {
39         $errMsg = "<p>Tallentamisessa jotain ongelmaa</p>";
40         return $errMsg;
41     }
42
43
44 }
```