



Episode 21 (Sales System V)



Tanuki, how about displaying the following figure on the browser for the purchaser of a rental CD and at the same time sending the same text to the purchaser's registered e-mail?

[Browser display of purchase acceptance]

The amount of money and products purchased by Shiro Saitama this time

Purchased Products : swan

Purchase Quantity : two sheets (pieces)

Amount billed or claimed : 500 Yen

Please transfer the funds to Tokyo ABC Bank within one week.

Account number :0523468 ordinary bank account

Account name : Used Equipment Sales Company Tokyo

Email sent.

[Email sent to purchaser]

受信箱 (合計 3通) - Evolution

注文をお受けしました!

差出人: master@
宛先: master@
件名: 注文をお受けしました!
日付: Sat, 30 Jan 2016 19:03:34 +0900 (JST)

今回の 埼玉 三郎 さんの購入金額と購入商品

購入商品: 白鳥

購入数量: 2 枚

請求金額: 500 円

1週間以内に東京ABC銀行にお振込をお願いします。

口座番号: 0523468 普通

口座名義: 中古販売会社東京

サーバから送信しています。返信しないでください。

master@s



Fox, I like it. Sounds interesting.
Does this mean that the admin is stuck at a PC and sending emails to each and every customer?



Tanuki, see the last sentence of the email above, "Sent from the server. Do not reply." This is not a lie, it is really being sent automatically from the mail server. This is why we have to add more mail servers to the server machine.

I'll give you a brief overview of the mail server, so Tanuki, as always, you'll have to build it on your own using the Internet.



I see, so I'm going to add my own mail server.
It sounds like a lot of work, but I'll give it a try.
Mail is for sending and receiving, right? Are there two types of mail servers?



Tanuki, you've got a good point. The outgoing server is called an SMTP (send protocol name) server, which requires the installation of a package called Postfix. However, it is often installed at the same time when CentOS7 is installed. If you find a configuration file named main.cf in the directory /etc/postfix/, you can assume that it is installed. Please check it.

The receiving server is called a POP (receive protocol name) server and requires the dovecot package to be installed.

Be sure to install this package. Both of them will function as a mail server.



I guess I should check the pros and cons of installing Postfix, and if not, install it, and then install dovecot. I have a strong ally, the Internet.



Tanuki, there are a lot of files that need to be set up after installation, so it's a lot of work.

In addition, you will need a mailer for testing, so get that ready as well. I have installed and use Tunderbird. You will need to be familiar with it before you can test the system.

I will give you a month to get used to it, so please do your best during that time.

After a month, I need you to work on the following.



Roger, roger that. I'm going to test my mettle.



It has been a month.

:

Tanuki, I hope you have finished building your mail server.

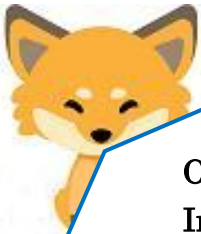
When you're done, we'll start the rest of the process.



I managed to finish.

I also tested sending and receiving emails to myself on Tunderbird.

I've created two email addresses, `raccoon@shns-m.com` and `fox@shns-m.com`.



Okay, let's move on.

In order to send mail automatically, you will use the `mb_send_mail()` function in the PHP script. To use this function, `php-bstring`, `php-imap` must be additionally installed to use this function.

In addition, `libc-client` has a dependency on `php-imap`, so you will have to install both.

Specifically, download the following two RPM packages, save them in the `Root`

directory and install them with the `rpm` command.

Search for and download `libc-client-2007f-16.el7.x86_64.rpm`.

Search for and download [https://rpmfind.net/linux/rpm2html/search.php?query=libc-client\(x86-64\)php-imap-5.4.16-9.el7.x86_64.rpm](https://rpmfind.net/linux/rpm2html/search.php?query=libc-client(x86-64)php-imap-5.4.16-9.el7.x86_64.rpm)

[https://rpmfind.net/linux/rpm2html/search.php?query=php-imap\(x86-64\)](https://rpmfind.net/linux/rpm2html/search.php?query=php-imap(x86-64))

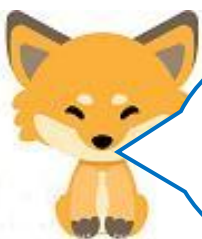
After saving the above two packages in the root directory, open a terminal and execute the following

```
# rpm -ihv libc-client-2007f-16.el7.x86_64.rpm  
php-imap-5.4.16-9.el7.x86_64.rpm
```



The two packages have to be imported from the outside, or the PC has to be connected to the Internet, which seems like a lot of work.

You're also new to the `rpm` instructions, used in the terminal. Is `-ihv` optional?



Option `-ihv` should be thought of as a spell for first-time installations.

However, you have to create a new table (`duserm`) in the database (`cdfox`) before creating the PHP program.

Because you would need an item to remember the customer's email address. I'll provide a sample below.

[Examples showing the creation]

Add a field (item) to the table (cduser) in the mysql database (cdfox) to store the email address named userMail, and name the table cduserm.

[Created]

```
MariaDB [cdfox]> create table cduserm(ucode varchar(4), uname varchar(100),  
userMail varchar(50), rireki varchar(4),kazu int);
```

[Creation Confirmation]

```
MariaDB [cdfox]> show columns from cduserm ;
```

Field	Type	Null	Key	Default	Extra
ucode	varchar(4)	YES		NULL	
uname	varchar(100)	YES		NULL	
userMail	varchar(50)	YES		NULL	
rireki	varchar(4)	YES		NULL	
kazu	int(11)	YES		NULL	

5 rows in set (0.01 sec)



The program to be executed is as follows. It will be a little big program, but since it includes both browser display and mail sending, it can't be helped.

The program name is [rtest036.php]. Naturally, the first program to run is rtest01.html, the next is rtest021.php, and the next is this rtest036.php. We need to modify the program to some extent.

```

<HTML>
<HEAD>
<TITLE> Register in database and notify customers</TITLE>
</HEAD>
<BODY>
<?php
$userCode = $_GET['userCode'];
$userName = $_GET['userName'];
$shoCode = $_GET['shoCode'];
$shoSus = $_GET['shoSus'];
$userMail = $_GET['userMail'];

$conid1 = mysqli_connect("localhost","ftpuser","ftpuser","cdfox");
if (mysqli_connect_errno()) {
    die("Unable to connect to database:" . mysqli_connect_error() . "¥n");
}
$query1 = "INSERT INTO cduserm (ucode, uname, userMail, rireki, kazu) VALUES
('$userCode','$userName', '$userMail', '$shoCode', '$shoSus')";
if (mysqli_query($conid1, $query1)) {
    echo " INSERT succeeded.¥n";
}
mysqli_close($conid1);
?>

<?php
$conid2 = mysqli_connect("localhost","ftpuser","ftpuser","cdfox");
$conid3 = mysqli_connect("localhost","ftpuser","ftpuser","cdfox");

$query2 = "select * from cduserm where ucode = '$userCode'";
// Execute the query.
if ($result2 = mysqli_query($conid2, $query2)) {
    foreach($result2 as $row2){
        $ucode = $row2['ucode'];
        $uname = $row2['uname'];
        $userMail = $row2['userMail'];
        $rireki = $row2['rireki'];
        $kazu = intval($row2['kazu']);
    }
}

```

```

    }
}
$query3 = "select * from cdshohin where scode = '$rireki'";
// Execute the query.
if ($result3 = mysqli_query($conid3, $query3)) {
    foreach($result3 as $row3){
        $scode = $row3['scode'];
        $sname = $row3['sname'];
        $jaketto = $row3['jaketto'];
        $music = $row3['music'];
        $zaiko = intval($row3['zaiko']);
        $kakaku = intval($row3['kakaku']);
    }
}
$gou = $kakaku * $kazu;
$zai = $zaiko - $kazu;
$query4 = "update cdshohin set zaiko = '$zai' where scode = '$scode'";
// Execute the query.
if (mysqli_query($conid2, $query4)) {
    echo " Update succeeded.¥n";
}
mysqli_close($conid2);
mysqli_close($conid3);
?>
<BR>
This time <?php echo $uname ? >'s purchase price and products purchased.<BR>
Purchased Products : <?php echo $sname; ?><BR>
Purchase Quantity : <?php echo $kazu; ?> bill<BR>
Amount billed or claimed : <?php echo $gou ?> Japanese Yen<BR><BR>
Please transfer the funds to Tokyo ABC Bank within one week<BR>
account number : 0523468    ordinary bank account<BR>
account name : Used Equipment Sales Company Tokyo<BR><BR>
<?php
mb_language('Ja');
$to = $userMail;
$from = "fox@shns-m.com";
$title = " We have accepted your order!";

```

```
$naiyou = " This $uname's Purchase Price and Items Purchased ¥n
Purchased Products : $sname ¥n
Purchase Quantity : $kazu bill ¥n
Amount billed or claimed : $gou Yen ¥n¥n
Please transfer the funds to Tokyo ABC Bank within one week.¥n
account number : 0523468 ordinary bank account ¥n
account name : Used Equipment Sales Company Tokyo ¥n¥n
Sent from the server. Please do not reply. ¥n$from ¥n";

if ($to != "" and $from != "" and $title != "" and $naiyou != "")
  {if (mb_send_mail($to,$title,$naiyou,"FROM:". $from))
    {echo " Email sent.";}
  else {echo " Failed to send email.";}
}
?>

</BODY>
</HTML>
```



The program is long and the content seems difficult.
What will be the result?



It is a bit difficult for a raccoon dog.
If done correctly, the result would look like this

[Example of display of rtest021.php modification]

Yahoo! JAPAN 検索結果
www

Music Listening
sazae-san

Customer Number (4-digit)
GK03

Customer name (25 characters)
Tokyo Saburo

E-mail address (50 characters)
raccoon@shns-m.com

Purchase item number (4 ch)
A002

Purchase quantity (single-byte)
3

Registration

[Example of browser display]

データベースへの登録と × +
保護されていない通信 | www

INSERT succeeded. Update succeeded.
The amount of money and products purchased by Shiro Saitama
this time
Purchased Products : swan
Purchase Quantity : two sheets (pieces)
Amount billed or claimed : 500 Yen
Please transfer the funds to Tokyo ABC Bank within one week.
Account number :0523468 ordinary bank account
Account name : Used Equipment Sales Company Tokyo
Email sent.

[Example of confirmation email display. Mailer is thunderbird]

We have accepted your order!

受信トレイ 注文をお受けしました ×

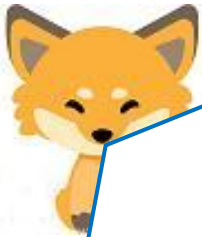
受信 作成 チャット アドレス帳 タグ

差出人 fox@shns-m.com 返信 転送 アーカイブ

件名 We have accepted your order!

宛先 (自分)★

The amount of money and products purchased by Shiro Saitama this time
Purchased Products : swan
Purchase Quantity : two sheets (pieces)
Amount billed or claimed : 500 Yen
Please transfer the funds to Tokyo ABC Bank within one week.
Account number :0523468 ordinary bank account
Account name : Used Equipment Sales Company Tokyo
Sent from the server. Please do not reply.
fox@shns-m.com



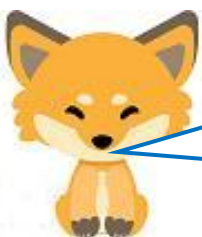
This is the end of the "sales system" story, although it was a bit long. From I to V, what we have talked about is exactly what is used in WordPress, which is provided by rental servers. HTML, JavaScript, and CSS are used for the user interface. PHP is used for the server-side scripting. The basic contents are from I to V. Naturally, MySQL is used as the database server. You cannot log in to MySQL from a terminal and manipulate the database directly, but you can use a tool (application) called phpMyAdmin to manipulate MySQL databases and tables, because it would be very dangerous if you accidentally destroy other users' databases.

That may seem like a lot of work to create an online store site with WordPress, but you can create a basic site, including database creation, with almost just a few clicks, so you should have no problem if you have one of the illustrated books. In addition, since the site theme and various plug-ins (programs such as PHP) are provided free of charge, you can create a site even if you have no knowledge of programming.

However, there is a big difference between having an overview from I to V in the head of a raccoon dog as back-up knowledge and not having it. Furthermore, knowledge of I through V is absolutely necessary when trying to customize themes and plug-ins.



Now I feel like I can use a rental server to create a WordPress site. Rental servers are cheap (about 500 yen/month), and it's safe and easy since I don't have to manage the server.



Let's start with **Episode 22**, and then we'll talk about security. We're going to take a little break from

Translated at DeepL