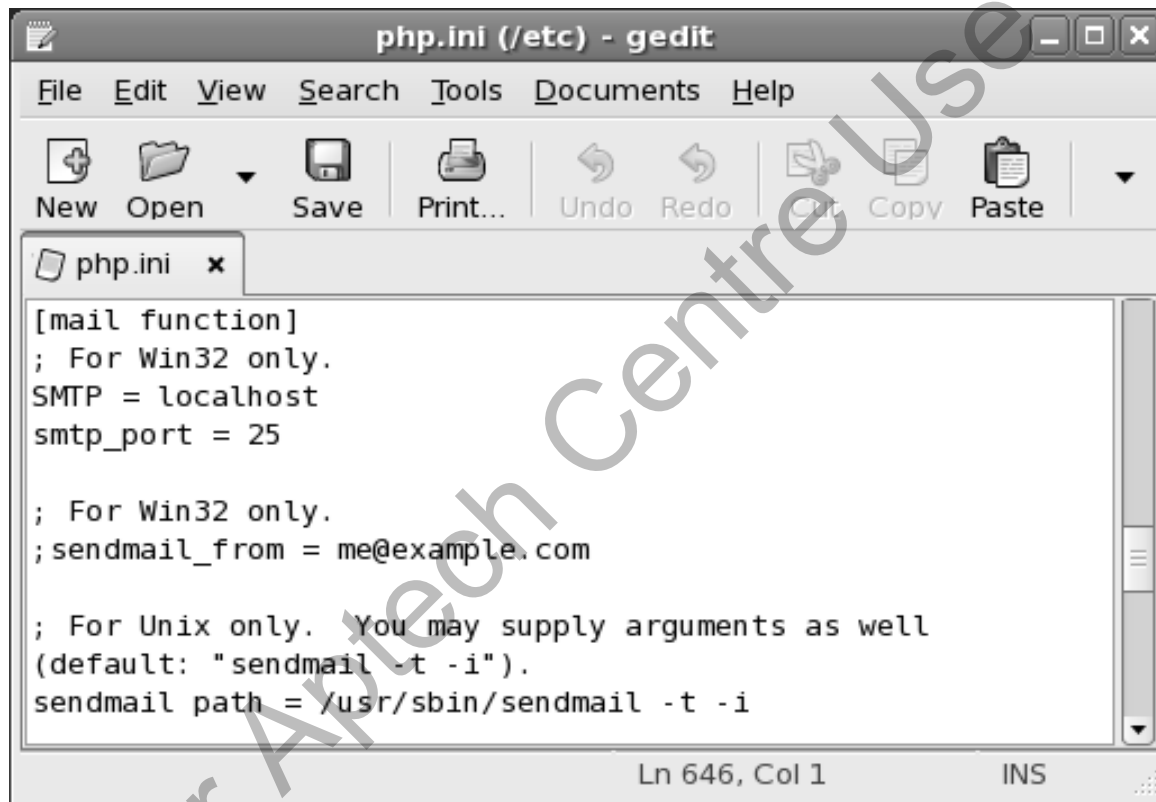# Handling E-mail with PHP

Session 19

◆ *Describe the process of sending an e-mail using PHP*

◆ *Explain the process of attaching files with an e-mail using PHP*

- E-mail are:
  - Fastest way of communicating with people across the world
  - It takes only few minutes to send and receive messages as compared to the usual hand written letters
- PHP provides the facility to send an e-mail

- In PHP, you can send an e-mail using the `mail()` function

- Location of the current local mail server must be specified in the `php.ini` configuration file to use the `mail()` function

- The `php.ini` file is located in the `/etc/` directory of the Linux operating system

The contents of the `php.ini` configuration are displayed as follows:

```
php.ini (/etc) - gedit

File   Edit   View   Search   Tools   Documents   Help

New   Open   Save   Print...   Undo   Redo   Cut   Copy   Paste

php.ini

[mail function]
; For Win32 only.
SMTP = localhost
smtp_port = 25

; For Win32 only.
;sendmail_from = me@example.com

; For Unix only.  You may supply arguments as well
(default: "sendmail -t -i").
sendmail path = /usr/sbin/sendmail -t -i

Ln 646, Col 1                    INS
```

◆ The `mail()` function is as follows:

Syntax

```
mail(to, subject, message , [additional_headers])
```

Where,

to - specifies the recipient's e-mail address

subject - specifies the subject for the e-mail

message - specifies the message that is to be written in the e-mail or the body of the e-mail

additional headers - specifies additional information such as the e-mail address of the sender and attachments

◆ **mail.php:** Sending an e-mail using the `mail()` function

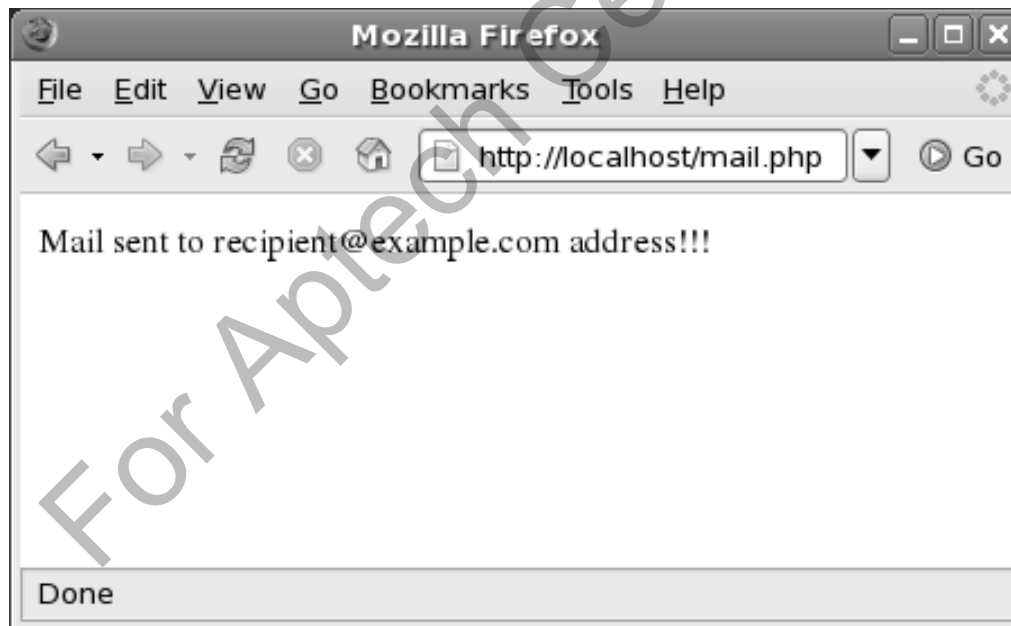### Snippet

```
<HTML>

<BODY>

<?php

$to = "recipient@example.com";

$from = "yourname@example.com";

$subject = "Test e-mail";

$body = "This is an example for showing the usage of
the mail() function.";

$send = mail($to, $subject, $body, $from);

if($send)

{

 echo "Mail sent to $to address!!!";

}
```

```
else

{

 echo "Mail could not be sent to $to address!!!";

}

?>

</BODY>

</HTML>
```

The following output is displayed:

- The `mail()` function is also used for sending mails to more than one recipient

- A `comma(,)` symbol is used for separating the e-mail addresses of the recipients

- **multi_rec.php** - Sending an e-mail to multiple recipients
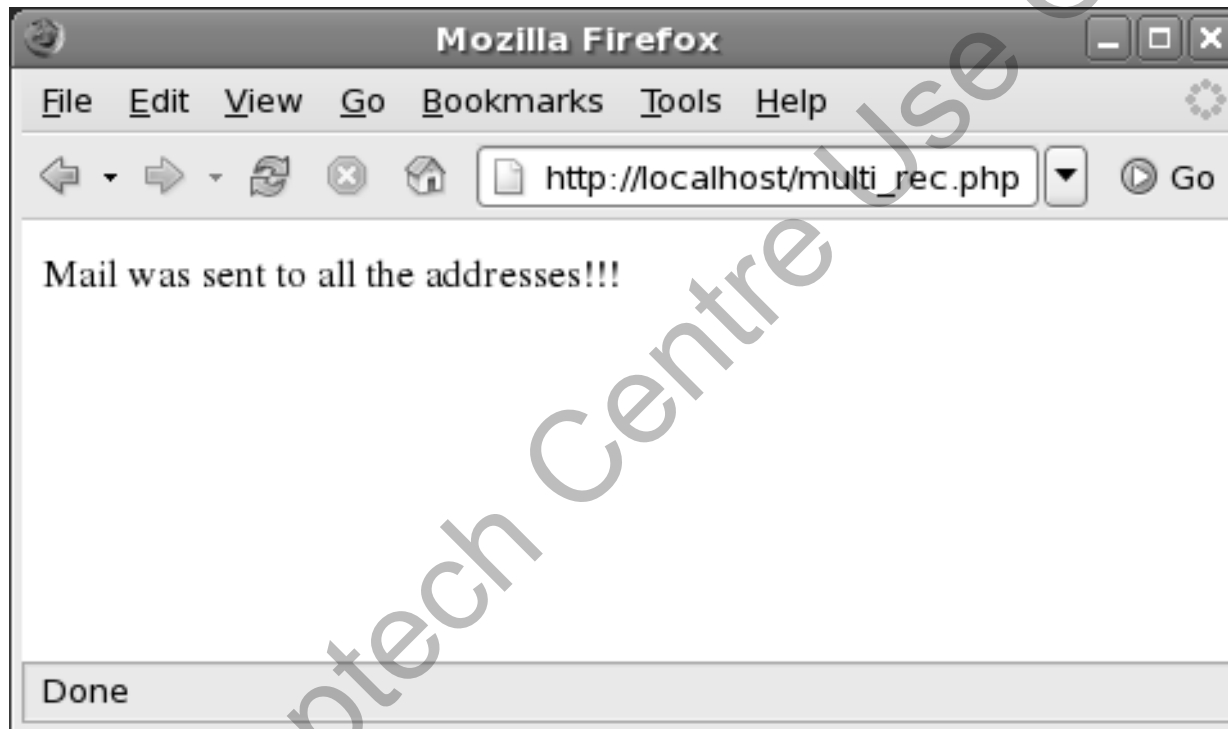
Snippet

```
<HTML>

<BODY>

<?php

$to = "recipient1@example.com, recipient2@example.com,

recipient3@example.com";

$from = "yourname@example.com";

for($i=0;$i<count($to);$i++)

{

 $to[$i] = trim($to[$i]);
```

```
$subject = "An example";

 $body = "This is an example for showing the usage of
                the mail() function.";

$send = mail($to, $subject, $body, $from);

if($send)

{

echo "Mail was sent to all the addresses!!!";

}

}

?>

</BODY>

</HTML>
```

The following output is displayed:

The `mail()` function is used to send an e-mail.

The `trim()` function is used to remove blank spaces.

◆ PHP also enables to read the e-mail addresses of the recipients from the text file and send an e-mail to them

◆ Steps for sending e-mail to multiple recipients whose addresses are stored in a text file are as follows:

1. Open a new file in the **gedit** text editor

2. Enter the following text:

```
recipient1@example.com

recipient2@example.com

recipient2@example.com
```

3. Save the file as **`email_list.txt`** in the `/usr/local/apache2/htdocs` **directory**

4. Assign read, write, and execute permissions to the **`email_list.txt`** file

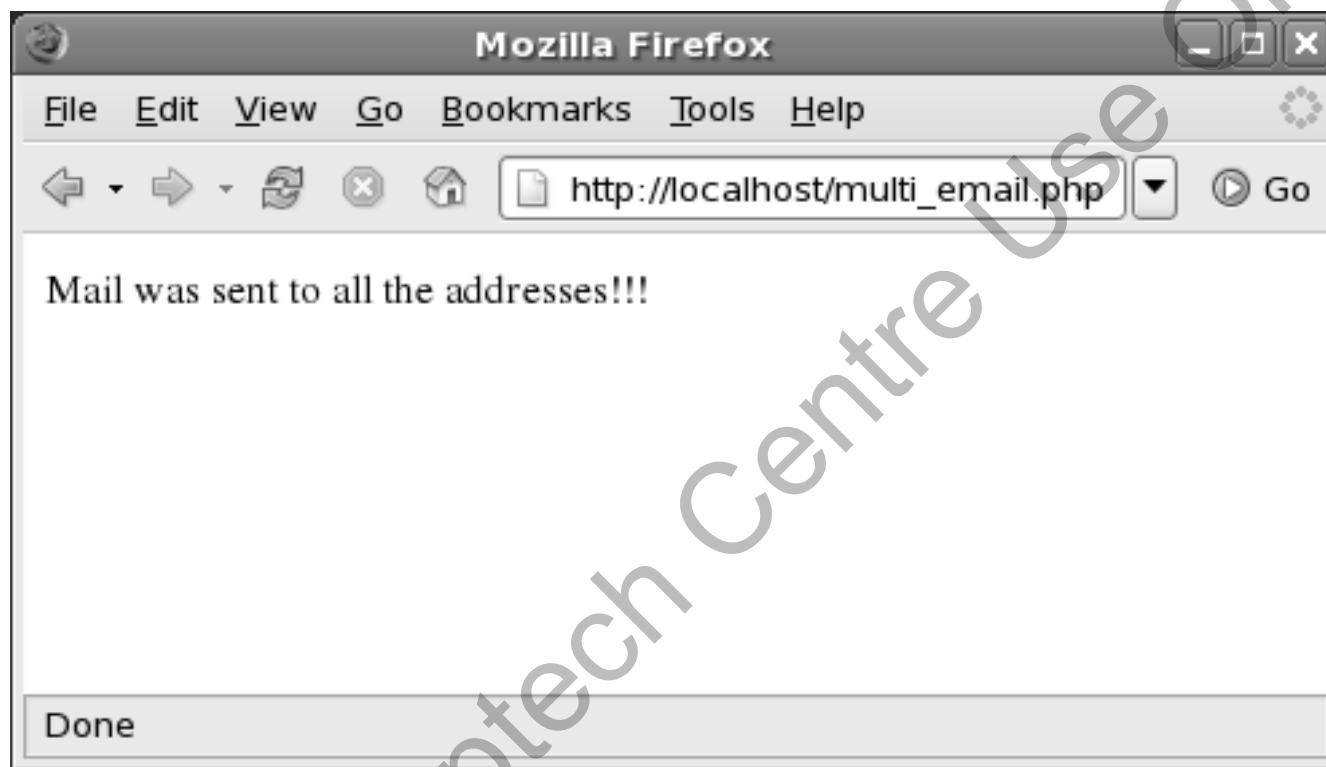5. Open a new file in the **gedit** text editor

6. Enter the code as shown in the Snippet:

Snippet

```
<HTML>
<BODY>
<?php
error_reporting(0);
$multi = "/usr/local/apache2/htdocs/email_list.txt";
$to_mail = file('$email_list.txt');
$from = "yourname@example.com";
for($i=0;$i<count($to_mail);$i++)
{
$to_mail[$i] = trim($to_mail[$i]);
$to = implode(",",$to_mail);
$subject = "An example";
$body = "This is an example for the mail() function.";
mail($to, $subject, $body, $from);
echo "Mail was sent to all the addresses!!!";
}
```

7. Save the file as **multi_email.php** in the `/usr/local/apache2/htdocs` directory

8. Open the Mozilla Firefox Web browser

9. Enter `http://localhost/multi_email.php` in the Address bar and press **Enter**

The following output is displayed:

The functions, such as `file()`, `trim()`, and `implode()` are used.

◆ ## An attachment is

- ◈ Any file that you want to send along with the e-mail

- ◈ Attached with the e-mail by setting up the header

◆ To send an e-mail with an attachment, perform the following steps:

1. Open a new file in the **gedit** text editor

2. Enter the following text:

   This is a sample attachment file

3. Save the file as **`example.txt`** in the `/usr/local/apache2/htdocs` **directory**

4. Assign read, write, and execute permissions to the **`example.txt`** file

5. Open a new file in the **gedit** text editor

6. Enter the code as shown in the Snippet

Snippet

```php
<?php

// filenames to be sent as attachment

$files = "/usr/local/apache2/htdocs/example.txt";

// e-mail fields: to, from, subject, and so on

$to = "recipient@example.com";

$from = "yourname@example.com";

$subject ="Test e-mail";

$message = "Sample message";

$headers = "From: $from";

// boundary

$semi_rand = md5(time());

$mime_boundary = "==Multipart_Boundary_x{$semi_rand}x";
```

```
// headers for attachment
$headers .= "\nMIME-Version: 1.0\n" ."Content-Type: multipart/mixed;
\n" . " boundary=\"{$mime_boundary}\"";
// multipart boundary
$message = "This is an example for mail attachment in MIME format.
\n\n" . "--{$mime_boundary}\n" .
"Content-Type: text/plain; charset=\"iso-9959-1\"\n" .
$message . // "\n\n";
$message .= "--{$mime_boundary}\n";
// preparing attachments
$file = fopen($files,"rb");
$content = fread($file,filesize($files));
fclose($file);
$message .= "Content-Type: {\"application/octet-stream\"};\n" . "
name=\"$files\"\n" . "Content-Disposition: attachment;\n"
. " filename=\"$files\"\n" . $content . "\n\n";
$message .= "--{$mime_boundary}\n";
```
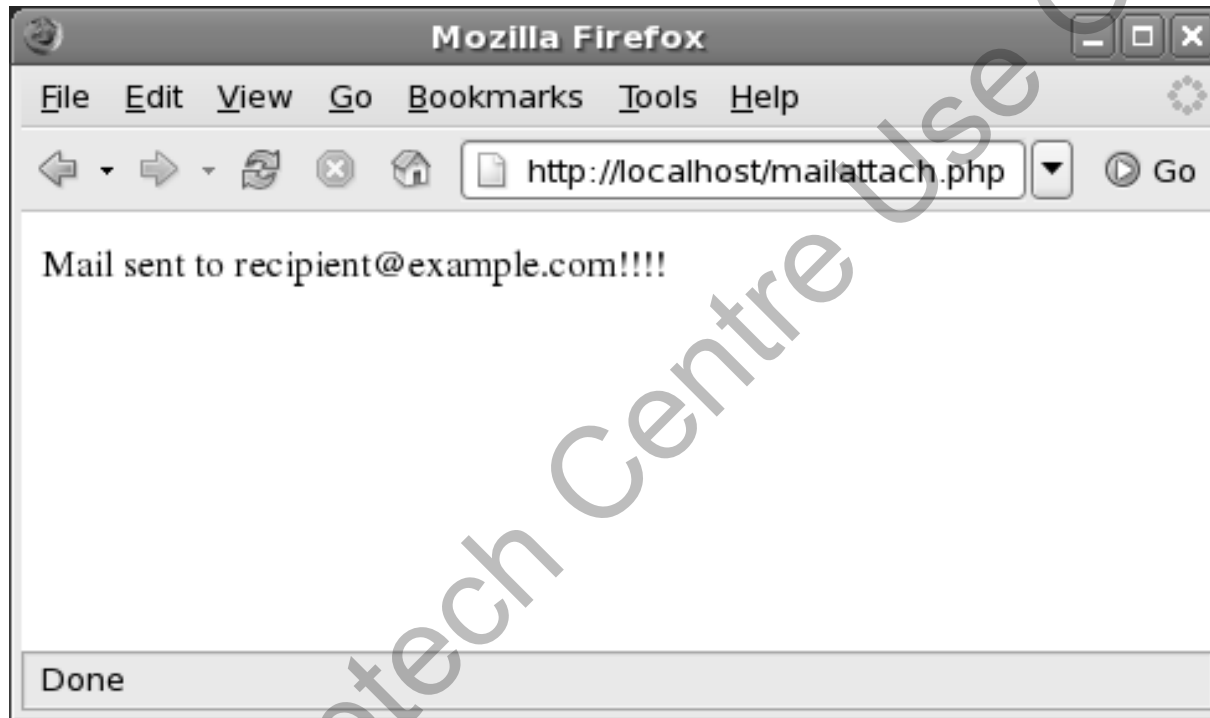
```
// send

$send_mail = @mail($to, $subject, $message,
$headers);

if ($send_mail)

{

echo "<p>Mail sent to $to!!!!</p>";

}

else

{

echo "<p>Mail could not be sent to $to!!!!</p>";

}

?>
```

7. Save the file as **`mailattach.php`** in the `/usr/local/apache2/htdocs` **directory**

8. Open the Mozilla Firefox Web browser

9. Enter **`http://localhost/mailattach.php`** in the Address bar and press **Enter**

The following output is displayed:

A boundary is created between the actual e-mail and the attachments.

The `fopen()` function opens the file that is attached with the e-mail in the read mode.

◆ The two built-in PHP function are as follows:

◈ `base64_encode()` - used for encoding the specified string

◈ `chunk_split()` - used for splitting the data into a series of smaller parts

◆ The mail() function is used for sending an e-mail

◆ The trim() function removes blank spaces included in between two recipient e-mail addresses

◆ The implode() function is used to join all the addresses present in the array with a comma operator

◆ An attachment can also be sent with an e-mail by setting up the header of the mail() function

◆ The uniqid() function assigns a unique identifier to the original mail and creates a boundary between the e-mail and the attachment

◆ The base64_encode() function is used for accepting the data in the form of a string and returns the data in the original form to the user

◆ The chunk_split() function is used for separating the data into smaller parts