Open Computing "Hands-On" Tutorial: October 1994: Listings

Listing 1: An HTML form, `order.html`, used by the fictitious Yoyodyne Corp. to collect order information via a graphical Web browser.

```
<TITLE>Yoyodyne Corp. Product Order Form</TITLE>

<H1>Product Order</H1>

<FORM ACTION="/cgi-bin/submit_order" METHOD="POST">

<UL>

<LI> Name: <P> <INPUT NAME="name" SIZE=80> <P>
<LI> Address:<P> <INPUT NAME="address" SIZE=80,3><P>
<LI> Email Address:<P> <INPUT NAME="email" SIZE=50><P>
<LI> Phone:<P> <INPUT NAME="phone" SIZE=30><P>
</UL>

<HR>

Items to order: <P>

<TEXTAREA NAME="items" ROWS=3 COLS=80></TEXTAREA> <P>

Choose method of payment: <P>
<SELECT NAME="payment">
<OPTION> Bill Me
<OPTION> Visa
<OPTION> American Express
<OPTION> Personal Check
<OPTION> Corporate Account
</SELECT>
<P>

If using a corporate account, please enter your account number and
authorization key: <P>

<UL>
<LI> Account Number<P> <INPUT NAME="account" SIZE=50><P>
<LI> Authorization Key<P> <INPUT TYPE="password" NAME="key" SIZE=50><P>
</UL>

<INPUT NAME="cc" TYPE = "checkbox" VALUE = "true"> Click here to send a
carbon-copy of your order to your email address (as specified above). <P>

<HR>

<INPUT TYPE = "submit" VALUE = "Submit Order"> <INPUT TYPE = "reset"
VALUE = "Clear Form"> <P>
</FORM>
```

Listing 2: The CGI program, `submit_order`, specified in `order.html`.

```
#!/usr/bin/perl
# Patrick Ryan (patrick.m.ryan@bell-atl.com)

eval "exec /usr/bin/perl -S $0 $*" if $running_under_some_shell;

# extra include directories
push(@INC,'/app/people/guide/GUIDE/server/WWW/httpd_1.1/cgi-bin');

require 'ctime.pl';
require 'cgi-lib.pl';

chop($now = &ctime(time()));

# Read in everything from httpd.
&ReadParse;

# Send the initial info back to the server.
print &PrintHeader;
print "<TITLE>Results of Your Order Submission</TITLE>\n";

unless ($in{'name'}) {
    print "You must include your name in any order.<P>";
    exit 0;
}

if ($in{'cc'} && !$in{'email'}) {
    print <<_EOT_;
You asked for a carbon copy of your order but did not include your
email address.  Please add your address and resubmit your order.<P>
_EOT_
    exit 0;
}

# Open up temporary file for ordering system
$order_file = "/tmp/order.$$";
open(ORDER,">$order_file");

print ORDER <<_EOT_;
NAME: $in{'name'}
ADDRESS:
$in{'address'}
PHONE: $in{'phone'}
EMAIL: $in{'email'}
PAYMENT: $in{'payment'}
ACCOUNT: $in{'account'}
AUTHORIZATION: $in{'key'}
_EOT_

print ORDER "ITEMS:\n";
$n=0;
foreach(split(/\n/,$in{'items'}))
{ ++$n; printf ORDER "%3d\t%s\n",$n,$_; }

print ORDER "RECEIVED: $now\n";
print ORDER "\n";
print ORDER <<_EOT_;
REMOTE: $ENV{'REMOTE_HOST'}
_EOT_
print ORDER "\n";
close ORDER;
```

```
# Send the order to the Processing database
$cmd = "/usr/local/bin/process_order $order_file";
system $cmd;

# If requested, send the user a copy of the order.
if ($in{'cc'} && $in{'email'})
{
    @addresses=($in{'email'});
    $to = join(' ',@addresses);
    # Escape any suspicious characters
    $to=&protect($to);

    $mail_cmd = "/bin/mail";
    $cmd = "$mail_cmd $to";

    unless (open(MAIL,"| $cmd")) {
        print <<_EOT_;
An error occurred while trying to submit your order.  Please contact
root@yoyodyne.com.<P>
_EOT_
    exit 0;
    }

    print MAIL "\n";
    open(ORDER,"<$order_file");
    while (<ORDER>)
    { print MAIL $_; }
    print MAIL "\n";

    close MAIL;
    close ORDER;

}

print <<_EOT_;
Thank you.  Your order was received at $now and has been sent to
the Processing Department.<P>
_EOT_

    unlink($order_file);
exit 0;

sub protect
# Quotify characters which are special to the shell
{
    local($_)=@_;
    s!([;:&\$'`|()])!\\$1!g;    # Use backslash to escape metacharacters
    $_;
}
```

Listing 3: Test script `cgi_test`.

```
#!/usr/bin/perl
#
# CGI script to process Software Modification Requests (SMRs)
# patrick.m.ryan@bell-atl.com (patrick m. ryan)
#
eval "exec /usr/bin/perl -S $0 $*" if $running_under_some_shell;
# extra include directories
```

```
push(@INC,'/app/people/guide/GUIDE/server/WWW/httpd_1.1/cgi-bin');
push(@INC,'/app/people/ryan/perl');

require 'cgi-lib.pl';
require 'date.pl';

$now = &date(time()-(4*3600));
select STDOUT;
$|=1;

# Suck in everything from httpd.
&ReadParse;

print &PrintHeader;
print "<TITLE>CGI test</TITLE>\n";

print &PrintVariables(%in);
print &PrintVariables(%ENV);
exit 0;
```