

Identification Division.
Program-id. cwb002.

this program is driven by the web browser and uses the CICS web and document API's. It accepts an account number from a web page and builds a response based on whether a record is found or not.
When a record is found, it builds a page dynamically in working-storage and then send it to the browser.
When the record is not found, an error page is sent to the browser.
When a CICS error is detected, a fatal error page is sent to the browser along with some other information.
The program is using 5 different templates:
The TEAM00ACCTEX2 template is used to display the account number menu. It does not need anything else and includes an HTML form.
The TEAM00TRAILEREX1 template is used to provide a link back to the exercise page; it should be inserted at the bottom of the document.
The FATALDOCUMENT template is used when we have a critical CICS error; the program name and date are added dynamically; the document should also insert the trailer template to complete this page.
The NOTFNDTEMPLATE template is used when a NOTFND condition is returned from the CICS READ command. It is self-contained and does not require anything else.
The TEAM00HEADER2 template is used to start the document that will return the account information to the browser; it only provides the header information on this screen.

Data division.
Working-storage section.

```
01 content-length      pic x(14) value 'Content-Length'.
01 content-value      pic x(80).
01 doc-token          pic x(16).
01 abs-time           pic s9(15) comp-3.
01 template-names.
   05 acct-template   pic x(48) value 'TEAM00ACCTEX2'.
   05 end-template    pic x(48) value 'TEAM00TRAILEREX1'.
   05 fatal-doc       pic x(48) value 'FATALDOCUMENT'.
   05 notfnd-doc      pic x(48) value 'NOTFNDTEMPLATE'.
   05 head-detail-temp pic x(48) value 'TEAM00HEADER2'.
```

Account-fields: data areas used when READ FORMFIELD issued

```
01 account-fields.
   05 accountno      pic x(8).
   05 accountlen     pic s9(8) comp value 8.
```

Account-first-list: symbol list used with the account template. The NEXTPGM field is used to tell the CICS web monitor program which application program to start when the response is sent back to us.

```
01 account-first-list.  
05          pic x(8) value 'nextpgm='.  
05          pic x(6) value 'CWB002'.  
05          pic x(9) value '&account='.  
  
01 account-field-names.  
05 account  pic x(7) value 'account'.  
05 next-program  pic x(7) value 'nextpgm'.
```

The next two "01" levels are used to describe the dynamic template added after the fatal template is inserted into the document and its list of symbols

```
01 fatal-details-temp.  
05          pic x(04) value '<ul>'.  
05          pic x(22) value '<li>program: &pgmname;'.  
05          pic x(22) value '<li>date: &datef;'.  
05          pic x(05) value '</ul>'.  
  
01 fatal-details-list.  
05          pic x(08) value 'pgmname='.  
05 pgmname  pic x(8) value spaces.  
05          pic x(07) value '&datef='.  
05 datef    pic x(10).
```

The next two "01" levels are used to describe the dynamic template used to display account detail information and its list of symbols

```
01 account-detail-temp.  
05          pic x(29) value '<basefont face="arial" size=+2>'.  
05          pic x(24) value '<ol><li>account number: '.  
05          pic x(8) value '&acctno;'.  
05          pic x(04) value '<br>'.  
05          pic x(19) value '<li>company: &cpny;'.  
05          pic x(18) value '<li>phone: &phone;'.  
05          pic x(22) value '<li>contact: &contact;'.  
05          pic x(05) value '</ol>'.
```

```

01 account-detail-list.
05          pic x(07) value 'acctno='.
05 acctno   pic x(8) value spaces.
05          pic x(6) value '&cpny='.
05 cpny     pic x(15) value spaces.
05          pic x(7) value '&phone='.
05 phone    pic x(10) value spaces.
05          pic x(9) value '&contact='.
05 contact  pic x(19).

01 FILE-BUFFER.
05 FL-RKEY  PIC X(8).
05 FL-CMPNAME PIC X(15).
05 FL-ADDR  PIC X(19).
05 FL-PHONE PIC X(10).
05 FL-BALANCE PIC X(8).
05 FL-RATING PIC X.
05 FL-CONTACT PIC X(19).

01 black-line      pic x(8) value '<hr><br>'.

```

Procedure division.

```

exec cics handle condition error(fatal-error) end-exec
exec cics asktime abstime(abs-time) end-exec
exec cics formattime abstime(abs-time)
                    ddmmyyyy(datef)
                    datesep('-')
end-exec

```

The "Content-Length" HTTP header tells us about the amount of data sent from the browser. A NOTFND condition indicates no data was sent. To us, it means we have to send the account inquiry form. This is found in the TEAM00ACCTEX2 template.

```

exec cics web read httpheader(content-length)
                    namelength(length of content-length)
                    value(content-value)
                    valuelength(length of content-value)
                    resp(eibresp)
end-exec

if eibresp = dfhresp(notfnd)
    continue
else
    perform display-details
end-if

```

At this point, we are about to send the account inquiry form. We will create a document using the account template. If any variables were to be present, they would have had to be set before executing the create command. In our case, it was done by priming working-storage with the proper information.

Content of TEAM00ACCTEX2 template:

```
<html>
<head>
<title>Untitled Document</title>
<meta http-equiv="Content-Type" content="text/html;
charset=iso-8859-1">
</head>

<body bgcolor="#FFFFFF" text="#000000">
<div align="center">
  <h3><font face="Arial, Helvetica, sans-serif" size="+2"
color="#0000FF">TEAM
  00</font></h3>
  <h4><font face="Arial, Helvetica, sans-serif" size="+2"
color="#0000FF">Module
  6 - Document exercise 2</font></h4>
</div>
<div align="center">
  <h4><font face="Arial, Helvetica, sans-serif" size="+1"
color="#0000FF">Account
  Inquiry</font> </h4>
  <form name="form1" method="post" action="&nextpgm;">
    <p align="left"><font face="Arial, Helvetica,
sans-serif">enter account number</font>:
      <input type="text" name="account" value="&account;"
maxlength="8">
    </p>
    <p>
      <input type="submit" name="Submit" value="Submit">
    </p>
    <p>&nbsp;</p>
  </form>
</div>
</body>
</html>
```

```
exec cics document create
  doctoken(doc-token)
  template(acct-template)
  symbollist(account-first-list)
  listlength(length of account-first-list)
end-exec
```

We are now sending the document - the CLNTCODEPAGE is important - it causes our data to be converted from ASCII to EBCDIC

```
exec cics web send doctoken(doc-token)
                        clntcodepage('iso-8859-1')
end-exec

exec cics return end-exec
.
```

display-details.

WE are here because we were able to read a Content-Length HTTP header; it tells us we have received an account number so a READ FORMFIELD is issued. This command would have to be used for each variable the program would have to get from the form

```
exec cics web read formfield(account)
                        namelength(length of account)
                        value(accountno)
                        valuelength(accountlen)
end-exec

exec cics read file('MASTERF')
                into(file-buffer)
                ridfld(accountno)
                resp(eibresp)
end-exec

if eibresp = dfhresp(normal)
    continue
else
    if eibresp = dfhresp(notfnd)
        perform record-not-found
    else
        perform fatal-error
    end-if
end-if

move fl-rkey to acctno
move fl-cmpname to cpny
move fl-phone to phone
move fl-contact to contact
```

The DOCUMENT CREATE command signals the beginning of our response to the account inquiry.

The TEAM00HEADER2 template contains the following information:

```
<html>
<head>
<title>Document exercise 2</title>
<meta http-equiv="Content-Type" content="text/html;
charset=iso-8859-1">
</head>

<body bgcolor="#FFFFFF" text="#000000">
<div align="center">
  <h3><font face="Arial, Helvetica, sans-serif" size="+2"
color="#0000FF">TEAM
    00 </font></h3>
  <h3><font face="Arial, Helvetica, sans-serif" size="+2"
color="#0000FF">Module
    6 - Document exercise 2</font></h3>
  <h4><font face="Arial, Helvetica, sans-serif" size="+1"
color="#0000FF">Account Inquiry</font>
  </h4>
<hr>
</div>
```

```
exec cics document create
      doctoken(doc-token)
      template(head-detail-temp)
end-exec
```

The DOCUMENT SET command adds a list of symbols to the document we just started. The DOCUMENT INSERT command inserts our working-storage template to the document.

Note that before inserting the template, all variables in the symbol list will have to be resolved.

```
exec cics document set
      doctoken(doc-token)
      symbollist(account-detail-list)
      length(length of account-detail-list)
end-exec

exec cics document insert
      doctoken(doc-token)
      from(account-detail-temp)
      length(length of account-detail-temp)
end-exec
```

```
exec cics document insert
      doctoken(doc-token)
      template(end-template)
end-exec
```

```
Content of the TEAM00TRAILEREX1 template:

<div align="center">
  <hr>
  <h3>&nbsp;</h3>
</div>
<div align="center"><font face="Arial, Helvetica,
sans-serif"><a href="team001.htm">Return
to exercise page</a></font></div>
</body>
</html>
```

```
exec cics web send doctoken(doc-token)
      cIntcodepage('iso-8859-1')
end-exec
```

```
The page we have sent should look like this:

      TEAM 00

      MODULE 6 - Document exercise 2

      Account Inquiry

-----

Account number: 00000100
Company: Longchamps Info
Phone: 4509673160
Contact: R Longchamps

-----

      Return to exercise page
```

```
exec cics return end-exec
fatal-error.

exec cics document create
      doctoken(doc-token)
      template(fatal-doc)
end-exec
```

We have just created the Fatal Error page using a template called FATALDOCUMENT; to this we have added some information from our program working-storage and the trailer template.

The content of the FATALDOCUMENT:

```
<html>
<head>
<title>Untitled Document</title>
<meta http-equiv="Content-Type" content="text/html;
charset=iso-8859-1">
</head>

<body bgcolor="#FFFFFF" text="#000000">
<div align="center">
  <h3><font face="Arial, Helvetica, sans-serif" size="+2"
color="#0000FF">Module
    6 - Document exercise 2</font></h3>
</div>
<div align="center">
  <h4><font face="Arial, Helvetica, sans-serif" size="+1"
color="#0000FF">Account
    Inquiry</font> </h4>
  <h4>&nbsp;</h4>
  <h5><font face="Arial, Helvetica, sans-serif" size="+1"
color="#FF0000">FATAL
    ERROR </font></h5>
  <p align="left"><font face="Arial, Helvetica,
sans-serif">A fatal error occured
    in a program running in the CICS server. Here is the
    information gathered
    at the time of error. An abend with a code of <font
color="#FF0000">TM00</font>
    was issued</font></p>
  <hr>
  <p align="left">&nbsp;</p>
</div>
```

```
move 'CWB002' to pgmname

exec cics document set
      doctoken(doc-token)
      symbollist(fatal-details-list)
      length(length of fatal-details-list)
end-exec

exec cics document insert
      doctoken(doc-token)
      from(fatal-details-temp)
      length(length of fatal-details-temp)
end-exec
```

```
exec cics document insert
  doctoken(doc-token)
  template(end-template)
end-exec

exec cics web send doctoken(doc-token)
                  cIntcodepage('iso-8859-1')
end-exec

exec cics dump dumpcode('tm00') end-exec

exec cics return end-exec
```

Record-not-found.

```
exec cics document create
  doctoken(doc-token)
  template(notfnd-doc)
end-exec
```

The NOTFNDTEMPLATE document template displays a message to the fact that the account entered was not found. It suggests to use the BACK button on the browser and to re-enter a new account number. Because our test environment does not have a very large file, we are also providing a table containing a list of all the valid keys.

```
exec cics web send doctoken(doc-token)
                  cIntcodepage('iso-8859-1')
end-exec

exec cics return end-exec
```