

# THE BRITISH COMPUTER SOCIETY

## THE BCS PROFESSIONAL EXAMINATIONS BCS Level 6 Professional Graduate Diploma in IT

### WEB ENGINEERING

23<sup>rd</sup> April 2008, 2.30 p.m.-5.30 p.m.

Answer THREE questions out of FIVE. All questions carry equal marks.

Time: THREE hours.

*The marks given in brackets are **indicative** of the weight given to each part of the question.*

Calculators are <b>NOT</b> allowed in this examination.
---

1. You are acting as a consultant for a financial institution, advising on the development of an online banking service.

- (a) One possible security risk to a banking site is **brute forcing** the password; this is often solved by locking out an account after a certain number of unsuccessful attempts.

- (i) What is meant by a brute force attack?

**(2 marks)**

- (ii) Aside from a brute force attack, outline **FOUR** other security risks and, for each risk, describe a method to prevent it.

**(8 marks)**

- (b) The in-house developers have outlined three possible schemes for authenticating their users. You have been asked to comment on these systems from a *security* perspective (how safe the scheme will be) as well as from a *user's* perspective (how easy the scheme will be to use).

**Proposed scheme A**

- Type in your *email address*
- Type in your *password* (6 characters, alphabetic)
- Select the first and the third digits of your *PIN* (6 digits) from a drop-down list

**Proposed scheme B**

- System-generated *user number* (created on registration, 11 digits long)
- Type in your *password* (6-12 alphanumeric characters)
- Type in your *mother's maiden name*

**Proposed scheme C**

- Type in your *first name*
- Type in your *last name*
- Type in your *postcode/Zip code*
- Type in your *date of birth*
- Type in your *favourite colour*

- (i) For each of the three schemes, outline **TWO** strengths and **TWO** weaknesses.

**(9 marks)**

- (ii) Devise a new scheme for authenticating users on this site which is superior to all three schemes outlined above. Explain how it overcomes the weaknesses identified in part (b)(i).

**(6 marks)**

**Turn over]**

2. A company selling digital cameras wishes to enhance its website to incorporate e-commerce. To this end, four modules are to be added to the existing website:
- User Registration & Login;
  - Shopping Cart (links into existing company financial and stock control databases);
  - Check Out (converting a *Shopping Cart* into an order, taking payment); and
  - Order Management (status of existing orders, cancelling pending orders, etc.).
- (a) (i) State **THREE** possible solutions to the problem of handling payments.  
(3 marks)
- (ii) Adopt **ONE** solution from part (i) and give a reason why it is preferable to each of the other solutions outlined.  
(2 marks)
- (b) In terms of the customer shopping experience on the website, outline the required functionality of the *Shopping Cart* module.  
(6 marks)
- (c) (i) Considering any **THREE** of the modules, identify the key aspects that would form a core part of an *acceptance test plan* for each. Amongst other things, the test plan should specifically address issues of performance and security.  
[Note: individual detailed test cases are not required here.]  
(10 marks)
- (ii) Construct an *integration test plan* involving all four modules.  
(4 marks)
3. (a) Write HTML to construct the form as indicated in Figure 3.1. When the button marked *Create Account* is pressed, the form should submit to a script called `create.php`  
(3 marks)
- (b) (i) What are the implications of *not* validating data?  
(2 marks)
- (ii) Describe, with an example for each, when data should be validated at:
- the client end; and
  - the server end.
- (3 marks)
- (c) Write code (either at the client side in JavaScript, or at the server side in ASP, PHP, or Perl) to validate that:
- (i) The `username` field exists and consists of 6 alphanumeric characters only.  
(3 marks)
- (ii) The `mobilephone` field is in the format “(07xxx) xxxxxx” where each “x” represents a single digit.  
(2 marks)
- (iii) The email field is in the format “<name>@<server>.<location>” (e.g. `invigilator-exams.room_5@bcs.org.uk`) where:
- (1) <name> consists of a combination of one or more alphanumeric characters, full stops (.), hyphens (-) and/or underscores (\_), e.g. “invigilator-exams.room\_5”.
  - (2) <server> is alphabetic only, e.g. “bcs”.
  - (3) <location> consists of a combination of one or more alphanumeric characters and/or full stops (.) e.g. “org.uk” or “com”.

The code should return an appropriate and informative error message if the validation fails.

**(3 marks)**

- (c) The server has a database named **ACCOUNTS**, with a single table named User (as shown in Figure 3.2). You may assume the web server and the database server are the same machine.

For parts (ii) and (iii) below, assume that the script `create.php` has already established a valid connection to this database. In all cases, you should state the language you are using (ASP, PHP or Perl).

- (i) Write code to connect to the database.

**(1 mark)**

- (ii) Write code to store the details submitted from the form into the database.

**(3 marks)**

- (iii) Write code to retrieve and display (in an appropriate format) the details of *all* accounts in the database which have an email address ending in **“.co.uk”**.

**(5 marks)**

*Note: The following SQL syntax may be useful to accomplish these tasks:*

- `INSERT INTO tbl_name (col1, col2, ...) VALUES (val1, val2, ...);`
- `SELECT * FROM tbl_name WHERE col1 like "%val1%";`

*(Where `tbl_name`, `col1`, `val1` etc. are to be replaced with appropriate values.)*

**Create new Account**

Enter a new username (6 characters):

Enter your mobile phone number:

Enter your email address:

Figure 3.1: HTML form

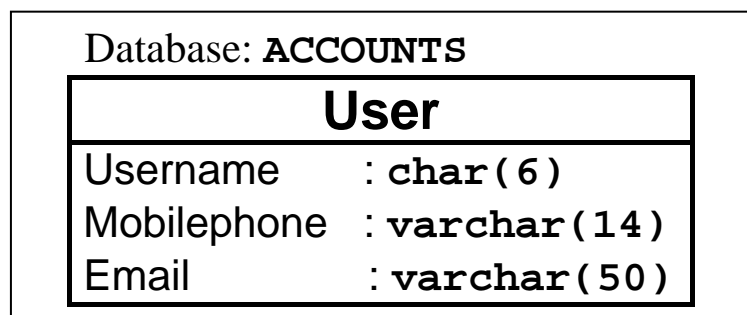


Figure 3.2: Database Entity-Attribute diagram

4. (a) (i) List the characteristics of **static** and **dynamic** web pages. (3 marks)
- (ii) Outline the technologies and tools used in the creation of a static web page containing images. (3 marks)
- (iii) Aside from those elements mentioned in (ii) above, outline the additional technologies and tools used in the creation of a dynamic web page. (4 marks)
- (b) Figure 4.3 (overleaf) details the source code of a web site for an electronics retailer.
- (i) Draw a diagram to complete the missing sections A, B and C indicated in Figure 4.1 (below) to illustrate the output of this file when it is first loaded in a browser window. (State the browser you are assuming use of.)(3 marks)
- (ii) The page has JavaScript actions attached to it. Describe what will happen on screen in relation to user interaction with the page. (4 marks)
- (iii) The retailer wishes to add extra details to the web page, with the same format and functionality as the current content. Write code to enable the menu group displayed in Figure 4.2 (below) to be generated and displayed in the same way as the other groups. (4 marks)
- (iv) Using CSS, list modifications to the code to change the appearance of the listed groups to the following:
- Cameras:** Bold text, surrounded by a solid 3-pixel border (2 marks)
- PCs:** Italic text, surrounded by a dotted 2-pixel border (2 marks)
- (Note: Your answer should list only the changes and additional lines required.)

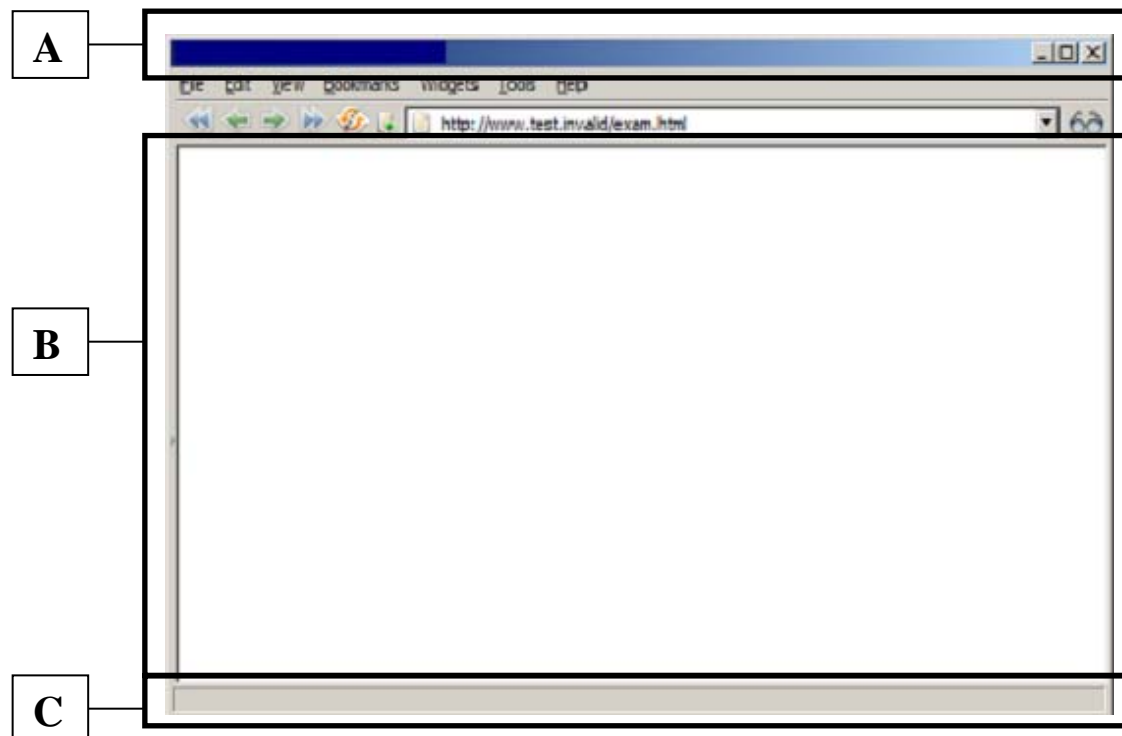


Figure 4.1: Template browser page for question 4



Figure 4.2: Menu group to add for question 4

Turn over]

```

1: <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
2:     "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
3: <html>
4:     <head>
5:         <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
6:         <title>Electronics Limited</title>
7:         <script type="text/javascript" language="javascript">
8:             <!--
9:             var activegroup = 0;
10:            var mingroupnum = 1;
11:            var maxgroupnum = 3;
12:
13:            function togglegroup(){
14:                activegroup++;
15:                if(activegroup > maxgroupnum) {
16:                    activegroup = mingroupnum;
17:                }
18:
19:            for(i = mingroupnum; i <= maxgroupnum; i++) {
20:                thisgroup = eval("document.all.group"+i+".style");
21:
22:                if(i == activegroup){
23:                    thisgroup.display = "block";
24:                }
25:                else {
26:                    thisgroup.display = "none";
27:                }
28:            }
29:        }
30:        -->
31:    </script>
32:    <style type="text/css">
33:        .group { display:none; margin-left:20px;}
34:    </style>
35: </head>
36: <body bgcolor="white">
37:     <h1 onmousedown="togglegroup()">Click here:</h1>
38:     <h3><a href="page1.html">
39:         Cameras</a></h3>
40:     <p id="group1" class="group">
41:         Sony<br />
42:         Canon
43:     </p>
44:     <h3><a href="page2.html">
45:         Televisions</a></h3>
46:     <p id="group2" class="group">
47:         Panasonic<br />
48:         Samsung
49:     </p>
50:     <h3><a href="page3.html">
51:         PCs</a></h3>
52:     <p id="group3" class="group">
53:         Dell<br />
54:         IBM
55:     </p>
56:     <p>
57:         <a href="http://validator.w3.org/check?uri=referer"></a>
60:     </p>
61: </body>
62: </html>

```

**Figure 4.3: Source code for question 4**

5. (a) A simple DTD for an academic journal is provided in figure 5.1.
- (i) Explain what it means for an XML document to be *well formed* and *valid*.  
(2 marks)
- (ii) The XML document in figure 5.2 contains precisely **FOUR** errors when validated against the DTD in figure 5.1. Identify all the errors, and provide a solution for each one.  
[Note: the line numbers are for your benefit and are not part of the XML code.]  
(4 marks)
- (iii) The original DTD has been found to be too simplistic for practical use. Referring to the XML document in figure 5.3, write a DTD which **enforces** the following **ten** constraints on JournalCatalogue:
- Sequence of elements is as shown in the XML code above (i.e. JournalCatalogue is a container of Journal elements).
  - The two attributes in Journal are mandatory.
  - Journal\_url and Editor\_email are optional.
  - Journal\_url must have a page attribute.
  - For each Journal, exactly one Journal\_title and one Editor element must be present.
  - For each Journal, the Article element must be present one or more times.
  - For each Article, exactly one Article\_title and one Abstract must be present and the Author element must be present one or more times.
  - An Author has a Name and optionally an Occupation.
  - A Name consists of both a First\_name and a Last\_name.
- (9 marks)
- (b) The website designers wish the XML list of journals to be displayed as a web page as indicated in figure 5.4. Using the HTML template provided in figure 5.5, complete the missing code in figure 5.5 (the section marked <!-- TO BE COMPLETED -->) to accomplish this.  
(10 marks)

```

1 : <?xml version = "1.0" encoding="ISO-8859-1"?>
2 : <!ELEMENT JournalCatalogue (Journal*)>
3 : <!ELEMENT Journal (Journal_title,Journal_url)>
4 : <!ATTLIST Journal ISBN CDATA "0-000-0000-0">
5 : <!ELEMENT Journal_title (#PCDATA)>
6 : <!ELEMENT Journal_url EMPTY>
7 : <!ATTLIST Journal_url page CDATA #REQUIRED>

```

Figure 5.1: XML simple DTD

```

1 : <?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
2 : <!DOCTYPE JournalCatalogue SYSTEM "journal-initial.dtd">
3 : <?xml-stylesheet type="text/xsl" href="journal.xsl"?>
4 : <JournalCatalogue>
5 : <Journal ISBN="0-123-4567-1">
6 : <Journal_Title>Nature</Journal_Title>
7 : <Journal_url>
8 : </Journal>
9 : </JournalCatalogue>

```

Figure 5.2: XML document with errors

```

1 : <?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
2 :     <!DOCTYPE JournalCatalogue SYSTEM "journal-ans.dtd">
3 :     <?xml-stylesheet type="text/xsl" href="journal-Cii.xsl"?>
4 : <JournalCatalogue>
5 :     <Journal ISBN="10104556" Issue_no="4">
6 :         <Journal_title>Nature</Journal_title>
7 :         <Journal_url page="http://www.nature.co.uk" />
8 :         <Editor>Bob Jones</Editor>
9 :         <Editor_email>bjones@whitensonz.com</Editor_email>
10:         <Article>
11:             <Article_title>Go green</Article_title>
12:             <Abstract>Global warming and eco systems</Abstract>
13:             <Author>
14:                 <Name>
15:                     <Last_name>Midori</Last_name>
16:                     <First_name>Vert</First_name>
17:                 </Name>
18:                 <Occupation>Academic</Occupation>
19:             </Author>
20:         </Article>
21:         <Article>
22:             <Article_title>See red</Article_title>
23:             <Abstract>Eco activism in developed countries</Abstract>
24:             <Author>
25:                 <Name>
26:                     <Last_name>Akai</Last_name>
27:                     <First_name>Rouge</First_name>
28:                 </Name>
29:                 <Occupation>Academic</Occupation>
30:             </Author>
31:         </Article>
32:     </Journal>
33:     <Journal ISBN="10104557" Issue_no="7">
34:         <Journal_title>Technology Today</Journal_title>
35:         <Editor>Michael Jackson</Editor>
36:         <Editor_email>jackmic@conv.com</Editor_email>
37:         <Article>
38:             <Article_title>Convergence</Article_title>
39:             <Abstract>One device as a container for many devices</Abstract>
40:             <Author>
41:                 <Name>
42:                     <Last_name>Gates</Last_name>
43:                     <First_name>Billy</First_name>
44:                 </Name>
45:                 <Occupation>Engineer</Occupation>
46:             </Author>
47:             <Author>
48:                 <Name>
49:                     <Last_name>Careers</Last_name>
50:                     <First_name>Steph</First_name>
51:                 </Name>
52:                 <Occupation>Academic</Occupation>
53:             </Author>
54:         </Article>
55:     </Journal>
56: </JournalCatalogue>

```

**Figure 5.3: XML sample document**



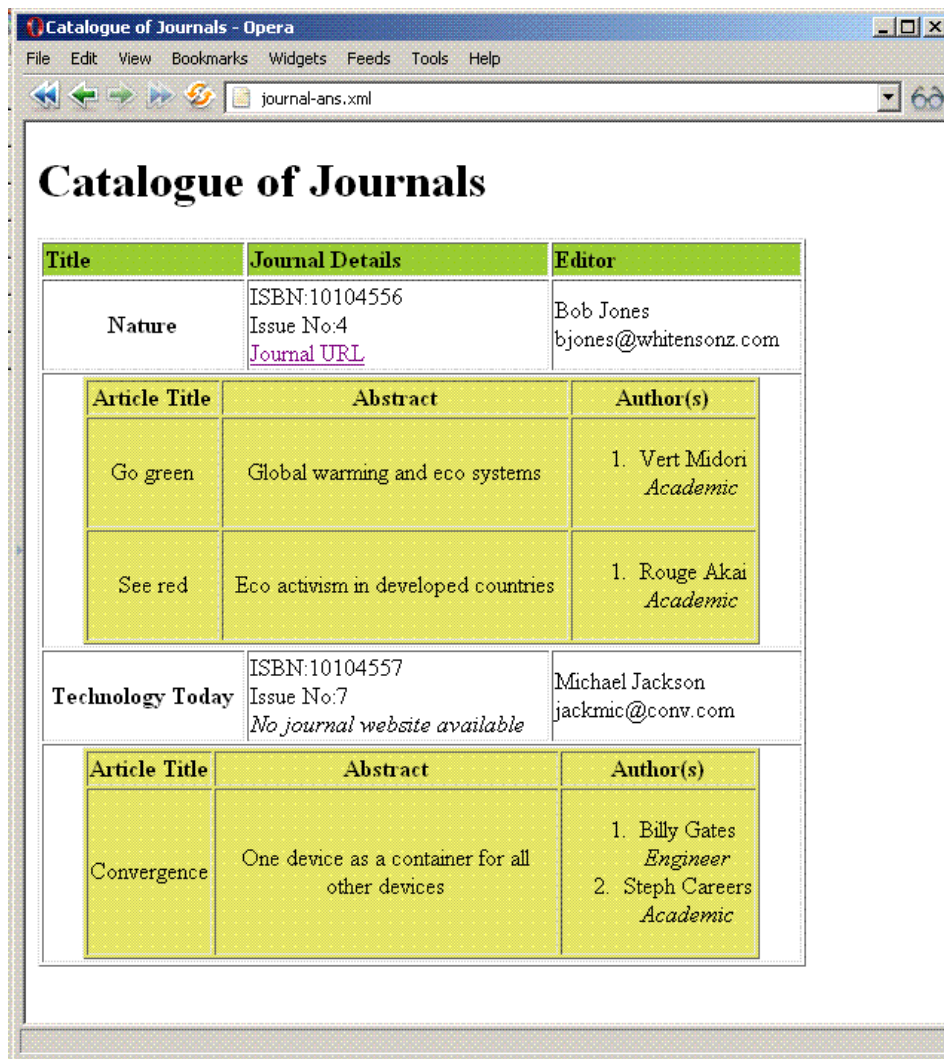


Figure 5.4: Sample browser page

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:template match="/">
    <html>
      <head>
        <title>Catalogue of Journals</title>
      </head>
      <body>
        <h1>Catalogue of Journals</h1>
        <table border="1">
          <tr bgcolor="#9acd32">
            <th align="left">Title</th>
            <th align="left">Journal Details</th>
            <th align="left">Editor</th>
          </tr>

          <!--TO BE COMPLETED -->

        </table>
      </body>
    </html>
  </xsl:template>
</xsl:stylesheet>

```

**Figure 5.5: Template HTML**