Mark Scheme (Results)

Summer 2023

Pearson Edexcel International Advanced Level
In Information Technology (WIT12) Paper 01

General Marking Guidance

- All candidates must receive the same treatment. Examiners must mark the first candidate in exactly the same way as they mark the last.
- Mark schemes should be applied positively. Candidates must be rewarded for what they have shown they can do rather than penalised for omissions.
- Examiners should mark according to the mark scheme not according to their perception of where the grade boundaries may lie.
- There is no ceiling on achievement. All marks on the mark scheme should be used appropriately.
- All the marks on the mark scheme are designed to be awarded. Examiners should always award full marks if deserved, i.e. if the answer matches the mark scheme. Examiners should also be prepared to award zero marks if the candidate's response is not worthy of credit according to the mark scheme.
- Where some judgement is required, mark schemes will provide the principles by which marks will be awarded and exemplification may be limited.
- When examiners are in doubt regarding the application of the mark scheme to a candidate's response, the team leader must be consulted.
- Crossed out work should be marked UNLESS the candidate has replaced it with an alternative response.

Question number	Answer	Additional guidance	Mark
1 (a)(i)	Award **one** mark for any of the following: • Specifies the version of HTML used (to build the web page) (1) • To ensure the web page is parsed the same way by different web browsers (1) • To let the browser know how the document should be interpreted (1)	Allow type of language	1
1 (a)(ii)	Award **one** mark for any of the following: • The value must be a single character/key / key on the keyboard (1) • There are too many characters (1)		1

Question number	Answer	Additional guidance	Mark
1 (b)	Award **one** mark for any of the following up to a maximum of **two** marks: • Starts on a new line (1) • There is a margin before and after the element / new lines are created before and after the element (1) • Takes up the full width available (1) • Can contain other block-level/inline elements (1)		2

Question number	Answer	Additional guidance	Mark
1 (d)	Award up to **two** marks for a linked explanation such as: • Functional testing assesses whether the web page works the way it should (1) whereas usability testing assesses how easy it is for a user to use the web page / (to complete the task)/ tested by others users (1)		2
		Total mark for question	**8**

Question number	Answer	Additional guidance	Mark
2 (a)	Award up to **two** marks for a linked explanation such as: • Rule 1 will find images/elements (with alt values) that begin with "bees"/a specific value (1) whereas Rule 2 will find images/elements (with alt values) that end with "bees"/a specific value (1) • ^ will apply to all images/elements (that have an alt value) that begins/is prefixed with "bees" (1) whereas $ will apply to all images/elements (that have an alt value) that ends/is suffixed with "bees" (1)	Does not require reference to ALT values	**2**

Question number	Answer	Additional guidance	Mark
2(b)	Award up to **two** marks for a linked description such as: • When the user hovers over the image (1) it will become half as/less opaque/0.5 (1) • When the user hovers over the image (1) it will become twice as/more transparent/0.5 (1)		**2**

Question number	Answer	Additional guidance	Mark
3 (a)(i)	Award **one** mark for any of the following: • t/the sixth character is not in the range a to r (1) • One or more uppercase characters is missing from the end of the input (1)		**1**

Question number	Answer	Additional guidance	Mark
3 (a)(ii)	Award **one** mark for: One example of an input that matches the regular expression R\d{2}[a-r]{3}(h|m)[A-Z]+ <table><tr><td>Uppercase R</td><td>R</td></tr><tr><td>2 digits</td><td>\d{2}</td></tr><tr><td>3 lowercase letters from a to r</td><td>[a-r]{3}</td></tr><tr><td>The character h or the character m</td><td>(h|m)</td></tr><tr><td>One or more uppercase characters</td><td>[A-Z]+</td></tr></table>		**1**

Question number	Answer	Additional guidance	Mark
3(b)	Award **one** mark for: [A-Z]{3}\d{2}[a-z]+(4|9)	Allow expanded repetition counts e.g. [A-Z][A-Z][A-Z] etc	**1**

Question number	Answer	Additional guidance	Mark
3 (e)	Award up to **two** marks for a linked explanation such as: • Keypress (event) will be triggered when the user presses a (character) key (1) whereas keyup (event) will be triggered when the user releases the key (1) • keypress indicates the character (1) whereas keyup indicates the key that was pressed (1)	Accept button instead of key	**2**
		Total mark for question	**12**

Question number	Indicative content	Mark

6	**Example of design principles**
	
	• visual hierarchy
	    ○ order in which a user processes information on a page should be taken into account
	    ○ F pattern is a reading pattern that resembles F
	    ○ Users generally read in an horizontal movement across the top, down and scan across a shorter width and then scan the left side
	• flow
	    ○ Guiding the user through the page
	• colour theory
	    ○ colour wheel
	    ○ monochromatic
	    ○ analogous
	    ○ complementary
	    ○ triadic
	    ○ compound
	• balance and contrast
	    ○ balance – symmetrical, asymmetrical, radial
	    ○ colours, shapes, space, textures etc.
	
	Consideration and significance
	• Most important things appear to be getting across
	    ○ there is a sports competition in July
	    ○ there are many events
	    ○ there is an opening and closing ceremony
	• Some attempt to use the F pattern e.g. title is top bar of the F
	• Scanning down and halfway across can pick up that it is mini Olympics and the events
	• No real attempt to capture the attention of the left side
	• Don't pick up much of the right-hand side without purposely moving attention there
	• Using blocks of colours and shapes to guide user to the flow. Easy to see the that the white blocks contain two different topics of information
	• Not easy to determine there is a rollover or what real purpose it serves. User would not know there was a rollover unless they accidently hovered there
	• Complementary colour scheme used. Opposite side of the wheel. Colours are effective
	• Asymmetric balance used for header and footer, symmetric used for content. Headings balance etc. Quite effective
	• Contrast used quite well taking into account the complementary colour scheme content is clearly distinguishable. Different sizes are used to make sure headings etc. stand out from content
	• Space between elements is quite good header to content, between columns etc.
		9

WIT12 - Unit 2 2023 - Code

Question number	Answer	Additional guidance	Mark
1 (c)	Award **one** mark for each of:		**2**
	C1 | `<h1>` tag closed `</h1>` (1)		
	C2 | `<ul>` tag added after `<li>Arrange for refreshments at 10 am</li>` and before `<li>coffee for 8</li>` (1)		
	Correct code is:		
| | ```html
<h1>To do list</h1>
<p>These are the tasks that need to be completed.</p>
<ul>
    <li>Morning meeting</li>
        <ul>
            <li>Set up meeting room</li>
                <ol>
                    <li>Desks arranged in circle</li>
                    <li>Resource pack printed for 15 people</li>
                </ol>
            <li>Arrange for refreshments at 10 am</li>
                <ul>
                    <li>Coffee for 8</li>
``` | | |

Question number	Answer	Additional guidance	Mark
2 (c)	Award **one** mark for each of:		3
	C1 shape centred horizontally (1)		
	C2 purple, blue and orange added as the gradients (1)		
	C3 transition timing of 3 seconds added (1)		
	Sample code		

```css
/* Add your code to centre the shape across the page */
margin: auto;

/* Add your code to set the colours of the shape */
background-image: linear-gradient(purple, blue, orange);

/* Add your code to set the transition time to 3 seconds */
transition: 3s;
```

Question number	Answer	Additional guidance	Mark
2 (d)	Award **one** mark for each of:		4

D1	scissor image rotated 90 degrees (1)
D2	dashed border added to coupon (1)
D3	h1 font size set to 50px (1)
D4	coupon number italicised (1)

Sample code

```
#scissors {
    width: 50px;
    /* Add your code here */
    transform: rotate(90deg);
}

#coupon {
    text-align: center;
    /* Add your code here */
    border-style: dashed;
}

h1 {
    /* Add your code here */
    font-size: 50px;
}

#couponNumber {
    /* Add your code here */
    font-style: italic;
}
```

Question number	Answer	Additional guidance	Mark
3 (c)	Award **one** mark for each of:		3
	C1 Condition that checks if the two numbers are equal (1)		
	C2 Condition that checks if num1 is higher than num2 (1)		
	C3 Message that num2 is higher than num1 / num1 is lower than num 2 / code for the display of the message (1)		
	Example code:		

```
if (num1 == num2) {
    message = "The numbers are the same."
} else if (num1 > num2) {
    message = num1 + "is higher than " + num2
} else {
    message = num2 + "is higher than " + num1
}
alert(message)
```

Question number	Answer	Additional guidance	Mark
3 (d)	Award **one** mark for each of:		**4**

D1	addName function is called (1)	
D2	The first name that has been input is stored in a variable (1)	
D3	The last and first names are added to the array in the correct order (1)	
D4	The names in the array are displayed (1)	

Example code:

```html
<!-- Amend the code to call the addName function -->
<input type="button" id="btnAdd" value="Add" onclick="addName()"></input>

    function addName()
    {
        lName = document.getElementById("lastName").value

        // Add your code to store the first name value
        fName = document.getElementById("firstName").value

        // Add your code to add the lastname and firstname to the array
        nameArray.push([lName,fName])
    }


    function displayNames()
    {
        nameArrayLength = nameArray.length
        for (count = 0; count < nameArrayLength; count++){
            // Add your code to display the lastname followed by the first name
            alert(nameArray[count][0] +" "+ nameArray[count][1])
        }
    }
```

Question number	Answer			Additional guidance	Mark
4	**Evidence found in**		Award **one** mark for each of:		**20**
	HTML – Head	A1	Uses an external style sheet	1	

	HTML – Body	A2	At least one HTML5 semantic element used to define part of the page:
			        `<header>`
			        `<nav>`
			        `<section>`
			        `<article>`
			        `<footer>`
				1	

	CSS	A3	Colour set appropriately for at least one of: • header background colour cadet blue #5F9EA0 • header text – dark slate blue #483D8B • content background colour – thistle #D8BFD8 • columns background colour – tana #D8D8BF • footer background colour cadet blue #5F9EA0 • footer text – dark slate blue #483D8B	1	
	CSS	A4	Font size set appropriately for at least one of: • **Everyone 4 Sports** 35 pixels • **Everyone 4 Sport July Competition** 40 pixels • **Something for Everyone** 25 pixels • hyperlink 12 pixels • Either of the column headings or final text 20 pixels • column text 16 pixels • footer text 16 pixels	1	
	CSS	A5	Alignment set appropriately for at least one of: • **Everyone 4 Sport July Competition** centre aligned	1	

			Something for Everyone centre aligned **Mini Olympics** centre aligned **Last Year's Competition** centre aligned - Column text is justified - Final text centre aligned		
	HTML – Body	A6	Hyperlink added to web page times.html	1	
	HTML – Body	A7	Hyperlink added jumps to the times paragraph on the web page times.html	1	
	CSS	A8	**Everyone 4 Sport** aligned to the right of the logo **(must see on the webpage)**	1	
	CSS	A9	Runner and football images have a width of 410 pixels	1	
	CSS	A10	Karate image has a width of 205 pixels	1	
	CSS	A11	Runner image changes to football image on hover **(must see on webpage)**	1	
	CSS	A12	Football image changes back to runner image when hover focus lost **(must see on webpage)**	1	
	CSS	A13	Email link added	1	
	CSS	A14	Suitable padding/margins used	1	

Award up to a maximum of **six** marks for the adherence to the component layout and the application of CSS using the levels based mark scheme below.					
Topic Area	0	1	2	3	Mark
Adherence to component layout design	No awardable content	There is little adherence to the component layout design, leading to a solution that is not fit for purpose or is not suitable for the intended audience.	An attempt to adhere to the component layout design leads to a solution that is, in parts, fit for purpose and is, in parts, suitable for the intended audience.	The webpage fully adheres to the component layout design and style requirements. The resulting solution is fit for purpose and is suitable for the intended audience.	3
Application of CSS to control presentation		There is little attempt to make use of the facilities of CSS to control appearance and style. Most components rely on default configuration.	An attempt has been made to use CSS to control the appearance and style of some components. This has been successful in some cases.	Consistent and accurate application of CSS is used throughout to control the appearance and style of all components.	3
				Total mark for question	**20**

Question number	Answer			Mark
			Award **one** mark for each of: **Marks**	**20**
5	**eventsResults.html**	A1	Athlete's name written to the table	1
		A2	Loop used (start at 0, stop at numRunners or length of array, increment)	1
		A3	Race time for the **current** athlete found **(has to also have A2)**	1
		A4	Race time for athlete compared against fastest time	1
		A5	Fastest time overwritten if race time is faster	1
		A6	One of the details of the **fastest** athlete found **(has to also have A2 and A3)** • athlete number • athlete (name) • school	1
		A7	One of details of an athlete displayed: • time • athlete number • athlete (name) • school	1
		A8	All of the details of the **fastest** athlete found and displayed **(has to also have A2, A3, A6)**	1
	Events.html	A9	Check to see which event type has been selected	1
		A10	Track – trackEvents combo is displayed **OR** Field fieldEvents combo is displayed	1
		A11	Track – fieldEvents combo is hidden **OR** Field – trackEvents combo is hidden	1

Award up to a maximum of **nine** marks for the Functionality, user interface design, use of notation. Using the levels based mark scheme below.					
Topic area	**0**	**1**	**2**	**3**	**Mark**
Appropriate functionality: • components and code have been decomposed into appropriate parts • dynamic behaviours are implemented in JavaScript.		The component parts of the program are incorrect or incomplete, providing a program of limited functionality that meets some of the given requirements. Mostly inappropriate logic used.	The component parts of the program are complete, providing a functional program that meets some of the given requirements. Some parts of the logic are clear and mostly appropriate to the problem.	The component parts of the program are complete, providing a functional program that fully meets the given requirements. The logic is clear and appropriate to the problem.	3
Appropriate interface design: • error messages and other status reports • Interactivity between JavaScript and HTML		The design of the user interface lacks consideration for fitness for purpose and the intended audience.	The design of the user interface is in parts fit for purpose and addresses some of the needs of the intended audience.	The design of the user interface is fully fit for purpose and suitable for the intended audience.	3

Topic area	0	1	2	3	Mark
Appropriate use of notation: • presentation of JavaScript ensures clear readability • comments provide clarity.		Uses programming constructs and techniques to produce some required outcomes in the code. Uses data types that are rarely appropriate to the problem. Limited use of accurate syntax. Limited appropriate use and manipulation of data structures. Some use of meaningful variable names with limited or unhelpful commenting. Parts of the code are clear and readable but much of it makes limited use of appropriate spacing and indentation.	Uses programming constructs and techniques to produce most of required outcomes in the code. Uses data types, some of which are appropriate to the problem. Uses mostly accurate syntax. Accesses and manipulates data structures to produce mostly correct results and/or outcomes. Uses mostly meaningful variable names, with some use of appropriate commenting. Code is mostly clear and readable, making some use of appropriate spacing and indentation.	Accurate programming constructs and techniques are used. Appropriate data types are selected. Accurate syntax is used. Data structures are accessed and manipulated efficiently. Meaningful variable names and comments are used throughout. Code is clear and readable, making effective use of appropriate spacing and indentation.	3
				Total mark for question	**20**

Event results

```
37 ∨          for (var row = 0; row < athletes.length; row ++)
38              {
39                document.write("<tr>")
40                // athleteNumber is written to the table
41                document.write("<td>"+ athletes[row][0] + "</td>")
42                // school is written to the table
43                document.write("<td>"+ athletes[row][1] + "</td>")
44
45                // Add your code to write the athlete's name to the table
46                document.write("<td>"+ athletes[row][2] + "</td>")
47
48                // time is written to the table
49                document.write("<td>"+ athletes[row][3] + "</td>")
50                document.write("</tr>")
51              }
52          </script>
```

c

```
76      // Add your code here to find the information for the fastest athlete
77      for (count = 0; count < numRunners; count ++)
78      {
79        raceTime = athletes[count][3]
80
81        if (raceTime < fastestTime)
82          {
83            fastestTime = raceTime
84            athleteNumber = athletes[count][0]
85            school = athletes[count][1]
86            athlete = athletes[count][2]
87          }
88      }
89
90      document.getElementById("fastest").value = fastestTime
91      // Add your code here to display the rest of the information for the fastest athlete
92      document.getElementById("athleteNum").value = athleteNumber
93      document.getElementById("athleteName").value = athlete
94      document.getElementById("athleteSchool").value = school
95
```

Event

```javascript
function getEvent()
{

  var event = document.getElementById("eventType");
  var optionChosen = event.options[event.selectedIndex].text;

  // Add your code here
  if (optionChosen == "Track")
  {
    document.getElementById("trackEvents").style.display = "block"
    document.getElementById("fieldEvents").style.display = "none"
  } else if (optionChosen == "Field")
  {
    document.getElementById("trackEvents").style.display = "none"
    document.getElementById("fieldEvents").style.display = "block"
  }
}
```