National 5
Coursework
Assessment Task

National 5 Computing Science
Assignment
Assessment task

This document provides information for teachers and lecturers about the coursework component of this course in terms of the skills, knowledge and understanding that are assessed. It must be read in conjunction with the course specification.

Valid for session 2023-24 only.

This assessment is given to centres in strictest confidence. You must keep it in a secure place until it is used.

This edition: January 2024 (version 1.1)

Contents

Version 1.1

Introduction

This document contains instructions for teachers and lecturers, and instructions for candidates for the National 5 Computing Science assignment. You must read it in conjunction with the course specification.

This assignment has 40 marks out of a total of 120 marks available for the course assessment.

This is one of two course assessment components. The other component is a question paper.

Instructions for teachers and lecturers

This assessment applies to the assignment for National 5 Computing Science for the academic session 2023–24.

The task is valid for 2023–24 only. Once complete, you must send the assignment responses to SQA to be marked.

You must conduct the assignment under a high degree of supervision and control. This means:

- candidates must be supervised throughout the session(s)
- candidates must not have access to e-mail or mobile phones
- candidates must complete their work independently — no group work is permitted
- candidates must not interact with each other
- with no interruption for targeted learning and teaching
- in a classroom environment

You can use any integrated development environments (IDE) that enables candidates to generate evidence — this includes online IDEs. However, the IDE must have a facility that prevents candidates accessing their files and tasks outside the supervised classroom environment.

Time

Candidates have 6 hours to carry out the assignment, starting at an appropriate point in the course, after all content has been delivered. It is not anticipated that this is a continuous 6-hour session, although it can be, but conducted over several shorter sessions. This is at your discretion.

You have a responsibility to manage candidates' work, distributing it at the beginning and collecting it in at the end of each session, and storing it securely in between. This activity does not count towards the total time permitted for candidates to complete the assignment.

Candidates are prompted to print their work at appropriate stages of the tasks. They can print on an ongoing basis or save their work and print it later. Whatever approach they take, time for printing is not part of the 6 hours permitted for the assignment.

Resources

Each candidate must have access to a computer system with a high-level (textual) programming language and **either**:

- database application or software that can create, edit and run SQL
- software that can create, edit and run HTML and CSS or HMTL.

This is an open-book assessment. Candidates can access resources such as programming manuals, class notes, textbooks and programs they have written throughout the course. These may be online resources.

You must not create learning and teaching tasks that make use of constructs required in the assessment task, **with the specific purpose of developing a solution that candidates can access during the assignment.**

You can provide candidates with templates, however these templates must only contain general starter code used in learning and teaching (for example, a web page that contains the HTML, title and body elements) — templates must not be tailored to this year's task.

There may be instances where restriction of network use is prohibited (for example, a local authority-managed network with specific limitations). However, it remains your professional responsibility to make every effort to meet the assessment conditions.

Reasonable assistance

The assignment consists of three independent tasks. They are designed in a way that does not require you to provide support to candidates, other than to ensure that they have access to the necessary resources. Candidates can complete the tasks in any order.

Once the assignment is complete, you must not return it to the candidate for further work to improve their mark. You must not provide feedback to candidates or offer an opinion on the perceived quality or completeness of the assignment response, at any stage.

You can provide reasonable assistance to support candidates with the following aspects of their assignments:

- printing, collating and labelling their evidence to ensure it is in the format specified by SQA
- ensuring candidates have all the materials and equipment required to complete the assignment — this includes any files provided by SQA
- ensuring candidates understand the conditions of assessment and any administrative arrangements around the submission and storage of evidence, and the provision of files
- technical support

Evidence

All candidate evidence (whether created manually or electronically) must be submitted to SQA in a paper-based format. The evidence checklist details all evidence to be gathered. You can use it to ensure you submit all evidence to SQA.

You should advise candidates that evidence, especially code, must be clear and legible. This is particularly important when pasting screenshots into a document.

There is no need for evidence to be printed single sided or in colour.

Alteration or adaptation

The tasks are in PDF and Word formats. Each task is available as a separate file from the secure site. Word files allow candidates to word process their responses to parts of the task.

You must not adapt the assignment in any way that changes the instructions to the candidate and/or the nature and content of the tasks. However, you can make changes to font size, type and colour and to the size of diagrams for candidates with different assessment needs, for example, visual impairment.

If you are concerned that any particular adaptation changes the nature and/or the content of the task, please contact our Assessment Arrangements Team for advice as soon possible at aarequests@sqa.org.uk.

Submission

Each page for submission has the number of the assignment task that it refers to, for example 1a, and contains space for candidates to complete their name and candidate number. Any other pages submitted, for example, prints of program listings or screenshots, must have this information added to them.

Specific instructions for teachers and lecturers: 2023–24

All candidates must complete task 1 (software design and development) and **either** task 2 (database design and development) **or** task 3 (web design and development).

It is at your discretion how you approach this optionality in assessment. The task your candidates complete might be pre-determined by your progress through the course, or you may be able to let candidates choose which task to complete.

You must follow these specific instructions and ensure that candidates are aware of what you will give them at each stage in the assessment.

Print each task on single-sided paper, where applicable:

- this allows candidates to refer to information on other pages
- this helps you manage tasks that are split into more than one part

Task 1 — part A requires candidates to identify input(s), process(es) and complete a design. They **must** submit their evidence for this part of the task to you before you issue part B.

Task 1 — part B is a separate section. This ensures that candidates are not able to access part A and change their responses.

Task 2 — part A requires candidates to complete an analysis and data dictionary. They **must** submit their evidence for this part of the task to you before you issue part B.

Task 2 — part B requires candidates to write and test queries using a populated database file.

A Microsoft Access file (DinoDiscoveries.accdb) is provided for candidates to use in part B. If your centre uses a different database management system, you can create the relational database using the CSV files or the text files provided.

If using the CSV files, you should set up all tables, fields and validation shown in the data dictionaries below. Referential integrity should also be enforced.

The text files contain SQL create and insert statements for each table. If you use the text files, you must add validation (shown in the data dictionaries below), appropriate for your version of SQL. Referential integrity should also be enforced.

Entity: Fossil					
Attribute name	**Key**	**Type**	**Size**	**Required**	**Validation**
fossilID	PK	Number		Y	
countryFound		Text	25	Y	
year		Number		Y	
person		Text	100	Y	
dinoID	FK	Text			Existing dinoID from dinosaur entity

Entity: Dinosaur					
Attribute name	**Key**	**Type**	**Size**	**Required**	**Validation**
dinoID	PK	Text	10	Y	
dinoName		Text	255	Y	
length		Number		Y	Range: >= 0.1 and <= 50
diet		Text	255	Y	
period		Text	10	Y	Restricted choice: Cretaceous, Jurassic, Triassic

Task 3 — part A requires candidates to design a web page. They **must** submit their evidence for this part of the task to you before you issue part B.

Task 3 — part B is a separate section. This ensures that candidates are not able to access part A and change their responses.

A folder titled 'WDD' is provided. This contains the CSS, HTML and media files candidates need to complete this task. These files must not be renamed, and they must remain in the folders provided.

Candidates **do not** need to print completed web pages in colour.

Instructions for candidates

This assessment applies to the assignment for National 5 Computing Science.

This assignment has 40 marks out of a total of 120 marks available for the course assessment.

It assesses the following skills, knowledge and understanding:

- applying aspects of computational thinking across a range of contexts
- analysing problems within computing science across a range of contemporary contexts
- designing, implementing, testing and evaluating digital solutions (including computer programs) to problems across a range of contemporary contexts
- demonstrating skills in computer programming
- applying computing science concepts and techniques to create solutions across a range of contexts

Your teacher or lecturer will let you know if there are any specific conditions for doing this assessment.

In this assessment, you have to complete two short practical tasks.

You must complete task 1 (software design and development) and **either** task 2 (database design and development) **or** task 3 (web design and development).

You may complete the tasks in any order.

Advice on how to plan your time

You have 6 hours to complete the assignment. Marks are allocated as follows:

- Task 1 — software design and development 25 marks (63% of total)

  AND EITHER
- Task 2 — database design and development 15 marks (37% of total)

  OR
- Task 3 — web design and development 15 marks (37% of total)

You can use this split as a guide when planning your time for each of the two tasks.

Advice on gathering evidence

As you complete each task, you must gather evidence as instructed.

Your evidence, especially code, must be clear and legible. This is particularly important when you paste screenshots into a document. You can print code from the software environment or copy and paste this into other packages such as notepad or Word.

Use the evidence checklist provided to make sure you submit everything necessary at the end of the assignment. Ensure your name and candidate number is included on all your evidence.

Evidence may take the form of printouts of code, screenshots, typed answers, hand-written answers or drawings of diagrams and designs.

Advice on assistance

This is an open-book assessment. This means that you can use:

- any classroom resource as a form of reference (for example programming manuals, class notes, and textbooks) — these may be online resources
- any files you have previously created throughout the course

The tasks are designed so you can complete them independently, without any support from your teacher or lecturer. This means that you:

- cannot ask how to complete any of the tasks
- cannot access any assignment files outside the classroom

Computing Science assessment task: evidence checklist

You should complete the checklist for task 1 and **either** task 2 or task 3.

Task 1 — software design and development

Evidence		Tick
1a	Completed task sheet showing input(s) and process(es)	
1b	Completed task sheet showing your running total design	
1c	Completed task sheet with test table showing the expected output	
1d	Printout of your program code and your program output	
1e	Completed task sheet with your evaluation	

Task 2 — database design and development

Evidence		Tick
2a	Completed task sheet showing the analysis of the inputs	
2b	Completed task sheet showing the completed data dictionary	
2c(i)	Printout of SQL statement to add new fossil discovery	
	Printout of the updated table	
2c(ii)	Printout of SQL statement to display names of the dinosaurs whose fossils were discovered after 1980 in the USA	
	Printout of the output from the SQL statement	
2c(iii)	Printout of SQL statement to update the details of the Aardonyx	
	Printout of the updated table	
2d	Completed task sheet stating reasons why the SQL statement does not produce the expected output	

Task 3 — web design and development

Evidence		Tick
3a	Completed task sheet with two functional requirements	
3b	Completed task sheet showing your wireframe design	
3c 3d 3e	Printout of edited code for janPromotion.html and style.css	
3f	Completed task sheet with description of two tests that could be performed on the website	
3g	Completed task sheet with your evaluation of the fitness for purpose of the 'janPromotion' page	

Please follow the steps below before handing your evidence to your teacher or lecturer:

- Check you have completed all parts of task 1 and either task 2 or 3.
- Label any printouts and screenshots with the task number (for example 1a, 2a).
- Clearly display your name and candidate number on each printout.

Task 1: software design and development (part A)

Delaney's Disco wants to provide training sessions for their DJs. They want a computer program to be developed to ensure each training session runs smoothly.

Program analysis:

The program will ask for the duration of a training session. The DJ enters the duration of each song they will play. The program will add up the duration of all songs until the total is greater than or equal to the duration of the training session. At this point, the program tells the DJ they have selected enough songs to complete the training session.

The program will select a random song when the DJ should switch on the foam machine.

Assumptions:

- The duration of the training session will be entered in minutes but stored as seconds.
- The duration of the training session will between 10 and 30 minutes.
- The duration of each song will be entered in seconds.
- When the final song is played in full, the training session duration may be longer than the original session duration entered by the user.

1a Complete the analysis below by identifying the input(s) and process(es).

(2 marks)

Input(s)
Process(es)
Output(s)
Display a message that enough songs have been selected to complete the training session. Display a message to switch on the foam machine.

Candidate name_____ Candidate number_____

1b Design the code that will use a running total to add up the duration of all songs and tell the DJ when they have selected enough songs to complete the training session.

You can use a flowchart, structure diagram or pseudocode design.

(3 marks)

- Check your answers carefully, as you cannot return to part A after you hand it in.
- When you are ready, hand part A to your teacher or lecturer and collect part B.

Candidate name_____ Candidate number_____

Task 1: software design and development (part B)

Program design (Pseudocode)

Main steps

1 initialise variables
2 get valid training session duration
3 convert duration of training session to seconds
4 calculate total duration of songs
5 display training session summary

Refinements

```
2.1   ask for duration of training session
2.2   while duration of training session < 10 or > 30
2.3       display error message to enter valid number
2.4       ask user to re-enter duration of training session
2.5   end while loop

3.1   duration of training session = duration of training session * 60

4.1   songCounter = 0
4.2   while total < duration of training session
4.3       get and store duration of next song in seconds
4.4       total = total + duration of next song
4.5       if total >= duration of training session
4.6           display message to inform user that they have entered enough songs
4.7       end if
4.8       add 1 to songCounter
4.9   end while loop

5.1   counter = 1
5.2   display message stating the number of songs played + songCounter
5.3   foamMachine = random number between 1 and songCounter
5.4   start fixed loop for each stored song duration
5.5       display counter + ":" + duration of next song
5.6       if foamMachine = counter
5.7           display message to start foam machine
5.8       end if
5.9       counter = counter + 1
5.10  end fixed loop
5.11  display total with message
```


1c Using the program design, complete the expected output in the test table below. Assume the foam machine message is displayed at song 2.

Type of test	User input	Expected output
Normal	Duration of training session: 15 song 1: 300 song 2: 300 song 3: 400	Enough songs have now be entered. The number of songs played was _____. 1: 300 _____ _____ _____ The total length of the training session was 1000.

(2 marks)

1d Implement the program in a language of your choice.

Ensure the program matches the design given and remember to test your program.

Print evidence of your program code.

(15 marks)

Candidate name_____ Candidate number_____

1e With reference to your code, evaluate your program by commenting on the following:

Efficient use of programming constructs	**(2 marks)**
Robustness of your completed program	**(1 mark)**

Candidate name_____ Candidate number_____

Task 2: database design and development (part A)

A group of dinosaur enthusiasts keep records of all dinosaur fossil discoveries. They want to use the information to teach young people about these amazing creatures.

Each dinosaur record contains the dinosaur's name, length (between 0.1 and 50 metres), diet and the period when the dinosaur existed (Cretaceous, Jurassic or Triassic).

The group also records information about each fossil discovery: the country where the fossil was found, the year discovered and the name of the person who recorded the fossil details.

The group wants to store this information in a database.

2a Complete the analysis of the inputs shown below:

(2 marks)

Dinosaur	**Fossil discovery**
dinoID dinoName length diet period	

Candidate name_____ Candidate number_____

2b Using the information provided, complete the data dictionary for the attributes: length, diet, period in the dinosaur entity. These attributes are required.

(3 marks)

Entity: Dinosaur					
Attribute name	**Key**	**Type**	**Size**	**Required**	**Validation**
dinoID	PK	Text	10	Y	
dinoName		Text	255	Y	
length					
diet					
period					

- Check your evidence carefully, as you cannot return to part A after you hand it in.
- When you are ready, hand in part A to your teacher or lecturer and collect part B.

Candidate name_____ Candidate number_____

Task 2: database design and development (part B)

2c Your teacher or lecturer will provide you with a completed database containing two linked tables.

(i) A new fossil has been discovered. Implement an SQL statement that will add the following fossil discovery to the database.

fossilID: 2061
countryFound: USA
year: 2023
person: Garath
dinoID: DINO_68

Print evidence of the SQL statement and evidence that clearly shows that the change has been implemented.

(2 marks)

(ii) The group is meeting youngsters from the USA. They want to show them the names of all the dinosaurs whose fossils were discovered after 1980 in the USA. The data should be sorted so that the earliest discovery is shown first.

Implement the SQL statement that will produce this list.

Print evidence of your SQL statement and the output from the query after it has been implemented.

(4 marks)

(iii) The details of the Aardonyx have been incorrectly entered. Implement an SQL statement that will update the record so that the length of the dinosaur is 8.0 metres and the diet is 'Herbivorous'.

Print evidence of the SQL statement and evidence that clearly shows that the change has been implemented.

(2 marks)

2d The following SQL statement is written to find the name, country and diet of all dinosaurs that were longer than 10 metres and whose fossils were discovered in the 1930s.

```
SELECT dinoName, country, diet
FROM Dinosaur, Fossil
WHERE year = 1930
AND length > 10;
```

Test this SQL statement and state two reasons why it does not produce the expected output.

(2 marks)

Reason 1
Reason 2

Candidate name_____ Candidate number_____

Task 3: web design and development (part A)

The 'Bonnie Wee Tattie' gift shop has a Scottish-themed toy on promotion each month. They want to add a new page to their website, with the following content for the January promotion:

- The name of this month's toy 'Hamish the Highland Coo' and the description.
- Information about the original price (£19.99) and sale price (£5.99).
- An image of the toy (height 250px by width 400px).
- An external link to the Scotland Wikipedia page (https://en.wikipedia.org/wiki/Scotland).
- A hyperlink back to the homepage.

3a Using the user requirements above, create two functional requirements for this web page.

(2 marks)

Functional requirement 1
Functional requirement 2

3b Your teacher or lecturer will provide you with a copy of the unfinished website.

Open this and look carefully at:

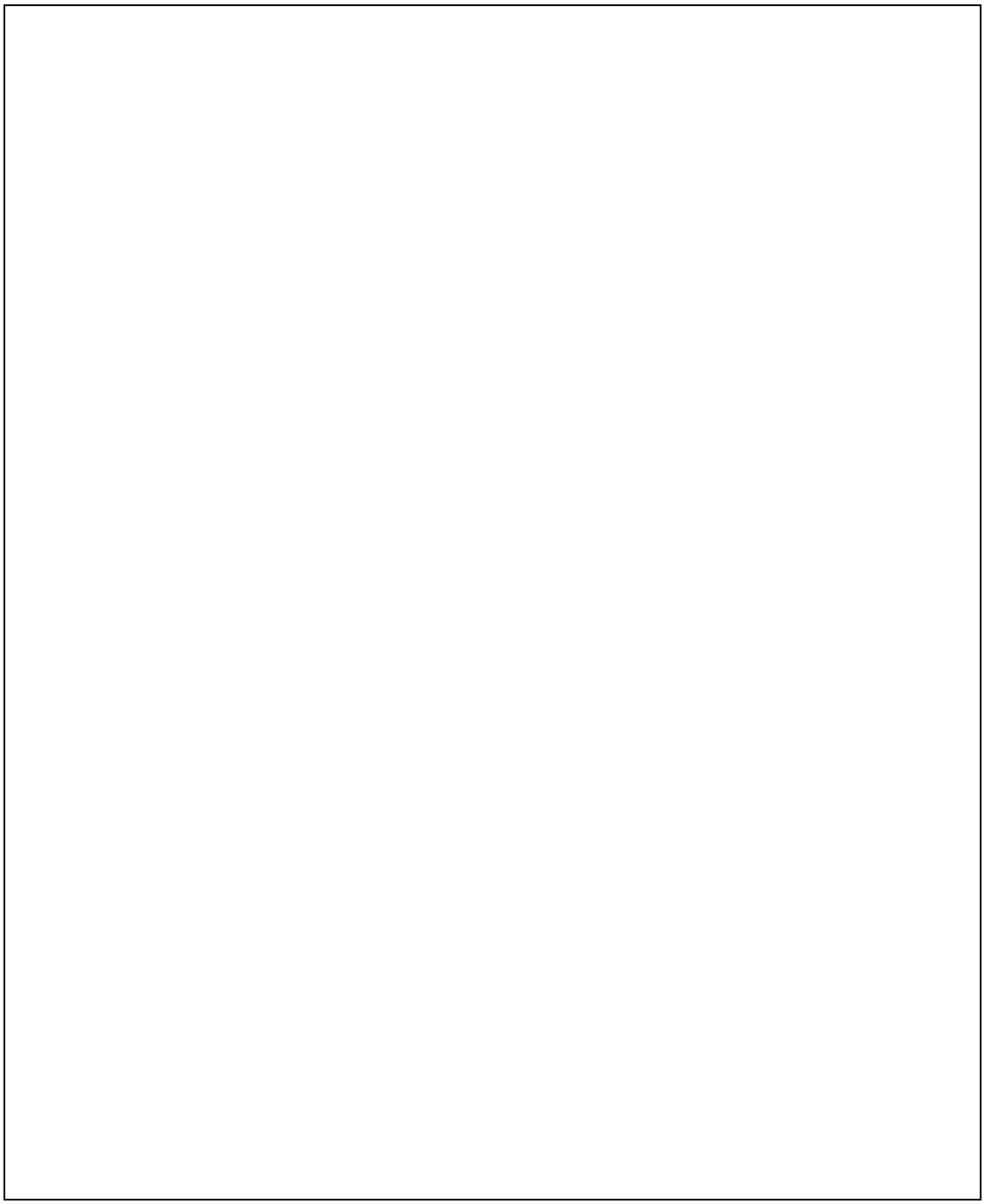
- the layout of each page
- the content of each page
- the navigation within the website
- any text and graphics that appear on every page

The 'janPromotion' page currently has no content. When complete, it should match the layout of the other pages.

Candidate name_____ Candidate number_____

Draw a wireframe design for the ‘janPromotion’ web page.

(2 marks)

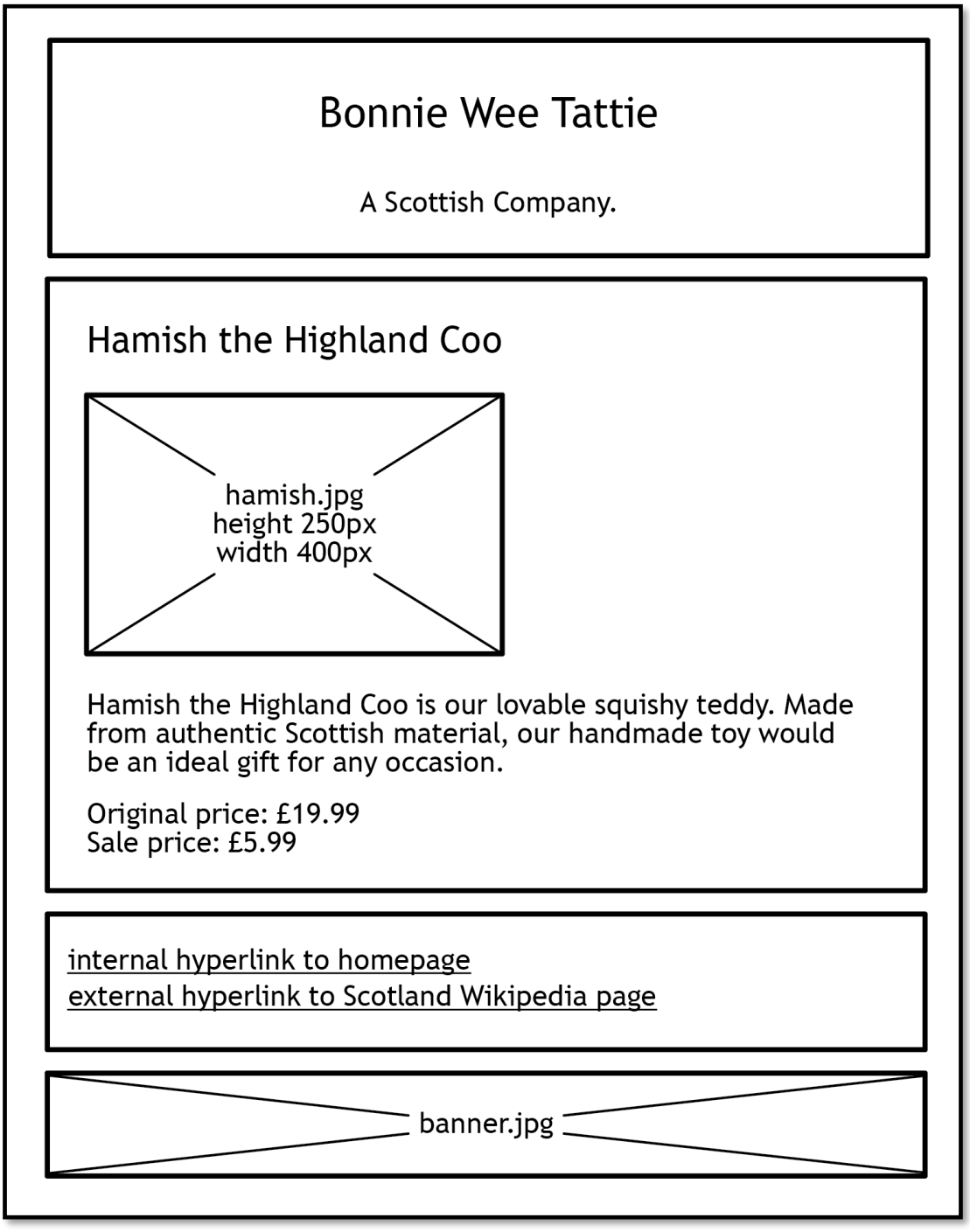


- Check your evidence carefully, as you cannot return to part A after you hand it in.
- When you are ready, hand in part A to your teacher or lecturer and collect part B.

Candidate name_____ Candidate number_____

Task 3: web design and development (part B)

A completed wireframe design for the janPromotions page is shown below.



Open the janPromotion.html and styles.css files in order to edit them.

3c Implement the design of the 'janPromotion' web page using HTML.

The file 'hamish.jpg' has been provided within the website files.

(4 marks)

3d The owners want to add a video clip of a real Highland cow to the 'janPromotion' web page.

Edit the 'janPromotion' web page so that it includes the video 'highlandCow.mp4' provided within the files.

(1 mark)

3e The owners decide they want to update the appearance of the website. Edit the code so that:

- the background colour of the content section and the background colour of the link section is: White (#FFFFFF)
- all hyperlinks are coloured: Bright orange (#FFAC1C)

On the 'janPromotion' web page all text should be:

- size: 18pt, with the exception of 'Sale price: £5.99' which should be size: 24pt
- text colour: Plum (#DDA0DD)
- text Alignment: centre

(3 marks)

Print evidence of your code from these edited files:

- janPromotion.html
- styles.css

3f Describe two tests that could be performed on this website.

(2 marks)

Test 1
Test 2

3g The functional requirements for the 'janPromotion' web page are:

- Display the name of the toy 'Hamish the Highland Coo'.
- Include the description of the toy.
- Display information about the original price (£19.99) and the sale price (£5.99).
- Display an image of the toy (height 250px by width 400px).
- Include an external link to the Scotland Wikipedia page.
- Include a hyperlink back to the home page.
- Show a video of a real Highland cow.

Evaluate the fitness for purpose of your completed 'janPromotion' web page.

(1 mark)

Candidate name_____ Candidate number_____

Copyright acknowledgements

Task 1 – martvisionlk/Shutterstock.com

Electronic files:

EVGEIIA/Shutterstock.com
VasiliyBudarin/Shutterstock.com
Proton-stock/Shutterstock.com
Svenkmajster/Shutterstock.com
William Barton/Shutterstock.com

Administrative information

Published: January 2024 (version 1.0)

History of changes

Version	Description of change	Date
1.1	Changes to task 3e: ♦ Removal of requirement for individual words to be different size ♦ Requirement for sale price moved and re-worded for clarity	24/01/24

Security and confidentiality

This document can be used by SQA approved centres for the assessment of National Courses and not for any other purpose.

This document may only be downloaded from SQA's designated secure website by authorised personnel.

