Cambridge International AS & A Level

INFORMATION TECHNOLOGY **9626/04**

Paper 4 Advanced Practical **October/November 2020**

MARK SCHEME

Maximum Mark: 110

Published

This document consists of **10** printed pages.

Generic Marking Principles

These general marking principles must be applied by all examiners when marking candidate answers. They should be applied alongside the specific content of the mark scheme or generic level descriptors for a question. Each question paper and mark scheme will also comply with these marking principles.

GENERIC MARKING PRINCIPLE 1:

Marks must be awarded in line with:

- the specific content of the mark scheme or the generic level descriptors for the question
- the specific skills defined in the mark scheme or in the generic level descriptors for the question
- the standard of response required by a candidate as exemplified by the standardisation scripts.

GENERIC MARKING PRINCIPLE 2:

Marks awarded are always **whole marks** (not half marks, or other fractions).

GENERIC MARKING PRINCIPLE 3:

Marks must be awarded **positively**:

- marks are awarded for correct/valid answers, as defined in the mark scheme. However, credit is given for valid answers which go beyond the scope of the syllabus and mark scheme, referring to your Team Leader as appropriate
- marks are awarded when candidates clearly demonstrate what they know and can do
- marks are not deducted for errors
- marks are not deducted for omissions
- answers should only be judged on the quality of spelling, punctuation and grammar when these features are specifically assessed by the question as indicated by the mark scheme. The meaning, however, should be unambiguous.

GENERIC MARKING PRINCIPLE 4:

Rules must be applied consistently, e.g. in situations where candidates have not followed instructions or in the application of generic level descriptors.
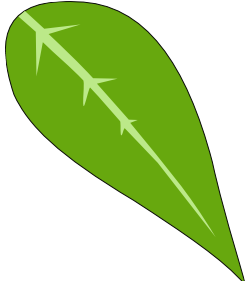

GENERIC MARKING PRINCIPLE 5:

Marks should be awarded using the full range of marks defined in the mark scheme for the question (however; the use of the full mark range may be limited according to the quality of the candidate responses seen).

GENERIC MARKING PRINCIPLE 6:

Marks awarded are based solely on the requirements as defined in the mark scheme. Marks should not be awarded with grade thresholds or grade descriptors in mind.

Task 1

Extract the 4 leaf cluster	Create a vector drawing of the leaf

4 leaf cluster clearly extracted		1
1 smooth unbroken side		1
2 smooth unbroken sides		1
3 smooth unbroken sides	With no shadow / ghosts and nothing behind	1
4 smooth unbroken sides		1
5 smooth unbroken sides		1
6 smooth unbroken sides		1
7 smooth unbroken sides		1
All (8) sides smooth unbroken nothing else to be seen		1
Central bud clear		1
Colour preserved	If sufficient background cut out	1
Resolution preserved		1
Transparent background		1

Non-Traced Vector graphics image of single leaf created	1
Vector version saved as svg	1
Non-Traced Bitmap version saved sized 50 px W or H	1
Simple even curves tear-drop shape – fat bottom – sharp tip	1
Colour matched	1
Central vein single smooth line	1
Central vein lighter colour	1
Central vein length **edge** to tip or 3/4 – not extended	1
Central vein tapers	1
Central vein has 2 or 3 light coloured barbs **...** allow 4th	1
... Central vein barbs correct shape 2 paired – all upwards	1
... Central vein barbs correct size reducing	1
... Central vein barbs correct position within 1st 3/4	1

Use leaf image and Branches file to match	

Can mark from an animation

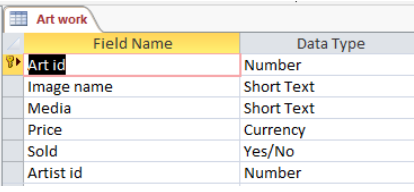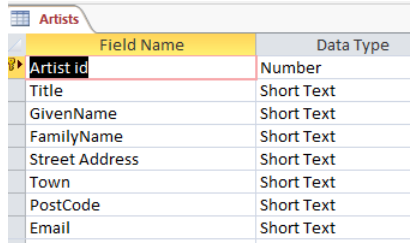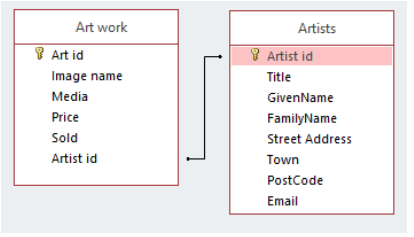|	
Branches image used and populated named Shrub – allow misspelling	1
⩾3 **clearly** different sized leaves	1
⩾3 leaves in **clearly** different orientations	1
At least 2 leaves behind branches	1
Lower right branch tip clear (in Shrub image)	1
Leaves on left branches correct positions and orientation	1
Leaves on central stem correct positions and orientation	1
Leaves on right branches correct positions and orientation	1
Transparent Bitmap Saved with size 400 × 400 1 px tolerance	1

Task 2 – **Animation**

Set frame 400 px by 400 px Insert leaf bitmap		Animate leaf Turn green to brown	
			Must change colour before falling unless it changes whilst falling or takes 1 sec at bottom
Frame size 400 × 400 Must play	1	Leaf changes colour **in stages** before falling	1
New (Their/populated) Shrub image inserted	1	Changes from green to brown, allow mottled	1
Image fits frame Consistent background, no clipping, all visible	1	Central vein changes colour	1
Their green Leaf image present on lower right branch	1	Central vein changes to yellow-brown	1
Leaf image inserted in correct position reaches very near the elbow	1	Change takes 1 second	1
Leaf image inserted with correct orientation	1	… Size does not change during animation	1
		… Shape is preserved	1

Animate brown leaf to fall to the bottom of the frame		
	Animation of leaf **falling** not jumping	1
	Descent is smooth	1
	… Descent is vertical only	1
	Descent takes 1 second	1
	Leaf stops at bottom of the frame/**tree** does not leave frame	1
	… Animation does not loop	1
	Animation saved as .gif or .swf and must animate	1
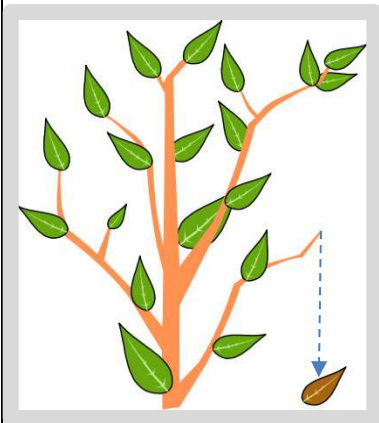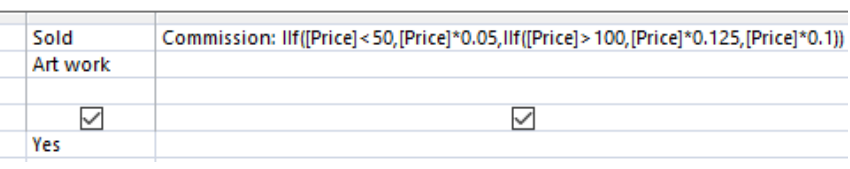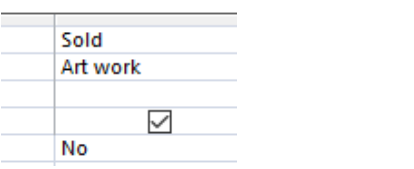
Task 3 – **Database**

Create a relational database using the ArtWork.csv and Artists.csv files.

ArtWork data correctly imported Record: ◄ ◄ 1 of 692 ► ►	1	Artists data correctly imported Record: ◄ ◄ 1 of 436	1	Relationship created on Artist id	1

ArtWork data correctly imported	1
Art id created as primary key	1
Price = currency	1
Sold = Y/N – boolean	1

Artists data correctly imported	1
Artist id created as primary key	1

Relationship created on Artist id	1

Generate data for the form

Sold	Commission: IIf([Price] < 50,[Price]*0.05,IIf([Price] > 100,[Price]*0.125,[Price]*0.1))	Sold
Art work		Art work
☑	☑	☑
Yes		No

Art **sold** query – sold criterion =yes,TRUE,–1	1	Art Unsold query created	1
Art sold results correct seen in query or table **15 records**	1	Correct criterion	1
Commission field added	1		
IIF operator used	1		
IIF criteria correct <50 >100 ⩽100	1		
Correct Commission Values seen in query or table	1		

> Can mark if **calculation** seen in form or table

Create a form to display the data

Form created	1
Title in serif font	1
Title text correct	1
Title colour correct	1

Artist fields correct	1
Artist fields correctly aligned	1

Artwork for Sale label correct	1
Artwork for Sale subform shown	1
Subform fields correct	1
Subform fields layout correct	1
Subform fields correct size	1

Artwork Sold label correct	1
Artwork Sold subform shown	1
Subform fields correct	1
Subform fields layout correct	1
Subform fields correct size	1

No scroll bars	1
No navigation bars in subforms	1
Navigation bar in main form	1
Data entry disabled	1

Task 4 – **JavaScript**

Insert JavaScript code to calculate commission.

Art Work Sales Commission Modelling

Enter Sale price boundaries

Sale prices below £ [50] will be charged 5% commission

Sale prices equal to this value and up to £ [100] will be charged 10% commission

Sale prices above this value will be charged 12.5% commission

Enter Artwork Price:

£ [30]

[Display Commission]

Commission for a sale of £30 will be £1.5

Art Work Sales Commission Modelling

Enter Sale price boundaries

Sale prices below £ [100] will be charged 5% commission

Sale prices equal to this value and up to £ [400] will be charged 10% commission

Sale prices above this value will be charged 12.5% commission

Enter Artwork Price:

£ [60]

[Display Commission]

Commission for a sale of £60 will be £3

50/100 – £30 entered = £1.5 displayed	1	100/400 – £60 entered = £3 displayed	1
50/100 – £50 entered = £5 displayed	1	100/400 – £100 entered = £10 displayed	1
50/100 – £100 entered = £10 displayed	1	100/400 – £300 entered = £30 displayed	1
50/100 – £120 entered = £15 displayed	1	100/400 – £400 entered = £40 displayed	1
		100/400 – £600 entered = £75 displayed	1
		Commission text displayed in position as shown in the question paper	1

```
<button type="button" onClick="Calculate()">Display Commission</button>
<br><!-- Placeholder for display inserted here -->
<h3 id="display here"></h3>
<SCRIPT>
//Function is code to run when button is clicked
function Calculate()
{
var low=LowerRange.value*1; //*1 to make variable numeric
var upper=UpperRange.value*1;
var price=ArtPrice.value*1;
var commission=0;
//Variables to contain boundary values, Art price and commission declared
if (price > upper) {commission = price*0.125;}
else if (price < low) {commission= ArtPrice.value*0.05;}
else {commission= price*0.1;}
//comparison of values to calculate the commission
document.getElementById("display here").innerHTML = "Commission for a sale of £"+price+" will be £"+commission
// Commission displayed at placeholder
}
</SCRIPT>

</body>
</html>
```

<SCRIPT> Tag inserted	1	Function inserted named "Calculate" or similar	1
</SCRIPT> Tag paired correctly	1	LowerRange id referenced in code	1
		UpperRange id referenced in code	1
<SCRIPT> Tags wholly within <head> or <body>	1	ArtPrice id referenced in code	1
		if – else if used (efficient method)	1
		Display placeholder(s) inserted id="any" under button	1
		// syntax used for comments inserted	1

[20]