GCSE COMPUTING CONTROLLED ASSESSMENT A452

By Bayley Cowen-Seagrove

RESEARCH TASK

CONTENTS

INTRODUCTION

I will gain my knowledge by searching via Google the code I do not understand. I will only use sites that are like w3schools and trustworthy companies. I will not find my information from sites such as Wikipedia, Wiki Answer etc. All the information I get from the internet will have a citation underneath. Another way I will research the code I do not understand is by articles in high quality books and magazines. I will ask a professional in the field to answer some of my questions about the validation of the form. I will not enter any personal information such as:

- Passwords
- Name in real life
- Address
- Age
- Bank details
- The school I attend

By not entering this data is ensures safety. When I collect information from away from a computer I will conduct myself in library and stores as I will represent myself. I will have to take breaks from looking at a computer screen and/or looking at a book/magazine/article as it can create nausea which can affect my state of mind. To keep my safety on a computer I will not download files from any un-trustworthy websites that state that they have files with the purpose of explaining in detail what I require as they could contain Malware, Viruses or programs that allow overtake of a computer.

DESCRIBE HOW DOES THIS HTML CODE PRODUCES THE FORM DISPLAYED IN THE BROWSER

The HTML code generates the code for the web browser.

HTML, Hyper Text Mark-Up, documents is another name for web pages that you see on your computer. It is a markup language that uses tags to describe the content of the document.

The title <title> Exam Entry <title> is what is seen on the tab in your web browser e.g. Google Chrome, Internet Explorer etc. This is the title of the page not the URL.

The JavaScript part of this document<script language="javascript" type="text/javascript"> begins when this code is entered in the form. This starts the script and identifies that the language that will be used in JavaScript.

THE FORM YOU SEE

<body> is used so everything after it can be seen on the form

The heading for the form is made by the code <h1>Exam Entry Form</h1>. This means that the name on the page will say 'Exam Entry Form' and will be in the font, heading 1.

The code that shows us the success page is formed by this code <form name="ExamEntry" method="post" action="success.html"> the form name is the name attribute specifies the name of a form. The method is post as this means after the click of the submit button the page is shown. The action is what will happen so for this code it will open a page called success.html and the code will take the document from your shared area. It shall only be successful if all data required is inputted and the code is right. If all data is entered and the page success.html does not show up either: the code is wrong, or you do not have a HTML document with the id success.html.

The textboxes are inside a table to keep the neat structure. The code <table width="50%" border"0"> creates a table that is not seen. It is not seen as the border is 0 so there is not a border that is visible. The width is 50% which means that the width of the table will be 50% of the page is what it will take up. The border is there not to see, as the border is <0, so it is only there to keep the data in an efficient order.

TAGS

The HTML tag <tr> and </tr> defines a row on a HTML table. The tag can contain two elements either <td> and <th>.

(w3schools, HTML tr tag)

The HTML tag <td> defines a standard cell in an HTML table. There are two types of tags and the <td> is the standard tag that contains data. The text in <td> elements are regular and left-aligned by default.

(w3schools, HTML td tag)(1)

MAKING THE DISPLAY (BUTTONS, NAMES)

The code <td id="name">Name</td> gives an id of 'name' to Name. The id is for naming particular objects in associated style sheets. The id should be unique to itself on the current document. If there are two objects with the same id on a page they are then seen as a collection and only can be accessed in ordinal order. The </td> ends the element

(Microsoft, 2014) (2)

The code <td><input type="text" name="name" /></td> tells us that the input type will be text so you must enter text for it to work.

The code <td id="subject">subject</td> gives an id of ''subject' to the value.

(Microsoft, 2014) (2)

The code <td><input type="text" name="subject" /></td> tells us that the input type will be text so you must enter text for it to work. The name of the input is what the name will be, for

this instance the name will be a subject. The </td> defines the end of the code of the standard cells in the html table.

To create the submit button you must use the code <td><input type="submit" name="Submit" value="Submit" onclick="return validateForm();" /></td>. The input type is submit, which means that the button will submit the form. The name is what you will see on the button so the name on this button is also Submit. The value is what the button will do so it will do to the data so this will submit the data. The code on click means when you click the button it will do the function without doing it when the page is opened.

To create the reset button to reset the inputs you have made you use the code <td><input type="reset" name="Reset" value="Reset" /></td>. The type is what it is for this it will reset so its function is to reset your input. The name is what you call the button. The value is what happens so it will reset the code. This is ended by the /td ending the code of the standard cells.

(w3schools, HTML td tag) (1)

DESCRIBE HOW THE JAVASCRIPT FUNCTION PERFORMS THE VALIDATION CHECK

FUNCTION

So the submit button can validate the form, we need to create a function called return validateForm so we use the code function validateForm() {. The brackets are left empty so the field/string is empty. A JavaScript function is a block of code designed to perform a particular task. A JavaScript function is executed when "something" invokes it. The validationForm validates if the field are valid so if the fields are meant to be not empty it will not validate the form.

(w3schools) (3)

VARIABLES

To create the variable to be true you use the code var result = true;. The variable is named result and the Boolean is true. The Boolean is if it has to make sense so for this it is true so the code must make sense. This means the variable is the result and it must make sense. If the Boolean was false the code will be not continue as if it's true it will.

(w3schools, Javascript Variables) (4)

The variable message var msg=""; has an empty field, string, to not allow any writing to happen. Having an empty field is different to having null or no text allowed (undefined) as having an empty field allows text but does not basically text with no given value.

(w3schools, Javascript Variables) (4)

IF THE FIELDS ARE EMPTY

To check that the fields are not empty the code if (document.ExamEntry.name.value=="") {. This means if the name value for form called Exam Entry is empty.

To make a message appear telling the user that they have left the name value as null the code msg+="You must enter your name \n"; The message will say 'You must enter your name' as when the form validates as it notices that the field is left empty and if the field is empty from the previous line of code, the message should appear. The \n is used to find a new line.

(w3schools, RegExp /n) (5)

The code document.ExamEntry.name.focus(); The focus method is used to give an element focus. This is giving the name focus.

(w3schools, Focus) (6)

The code document.getElementById('name').style.color="red" means that it will get the element by its id so will get the element by the id name. If the code is entered in wrong, the colour of the id will turn red to show to the user that that's the field they entered incorrectly, or didn't enter at all.

(w3schools, HTML DOM) (7)

The code result=false; is the same as event.preventDefault where if a callback occurs where it prevents the normal behavior so if the form was meant to be submitted and a callback occurred it would not submit.

To check that the field for subject is not empty if (document.ExamEntry.subject.value=="") { . This means that if the document, Exam Entry has an empty field for subject the following code should happen.

A message for the user that the field is empty is made by msg+="You must enter a subject \n";. This means a message will occur that will say that you must enter a subject as the field is empty. The /n is to start a new line.

(w3schools, RegExp /n) (5)

document.ExamEntry.subject.focus();. This method gives focus to the element subject. The focus() method triggers the focus event, or attaches a function to run when a focus event occurs.

(w3schools, jQuery Focus) (8)

document.getElementById('subject').style.color="red";. This means that in the document it will get the element with the ID of subject and make the colour will be red. This will only happen when the user does not enter their name or subject. If they enter their name, however fail to enter their subject only 'Subject' would turn red.

(w3schools, HTML DOM) (7)

The Boolean object result = false; is used to a non-Boolean value into a true or false (Boolean Value). As it is a Boolean value anything such as correct, incorrect, null will not be accepted as they are non-Boolean values; the true/false must be only used as they are Boolean values. If the result is false it will do return however if the result is true it will continue with the code.

(w3schools, JS Booleans) (9)

For the form to validate if the fields are empty the code if(msg==""){. This means if the field is empty or has null value.

The code return result; is used to not allow the code to continue if the field is empty. If the code is not empty the validation skips the warning as it is not needed. When JavaScript reach a return statement, the function will stop executing. If the function was invoked, the javascript will "return" and continue to execute the code.

(w3schools) (3)

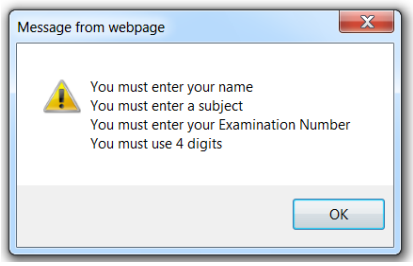
ALERT BOX

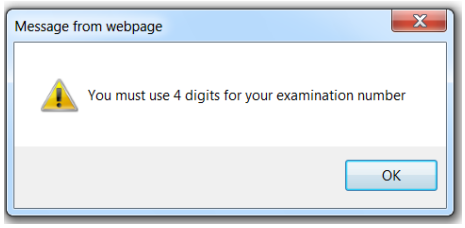
If the string/field is left empty a message has to warn the user that they have not sufficiently entered the data needed. An alert box is often used if you want to make sure information comes through to the user. The code alert(msg)is used. Earlier in the code a message for both empty name field and subject field was created. This creates an alert box and sends the message if they have either left one field empty or both.

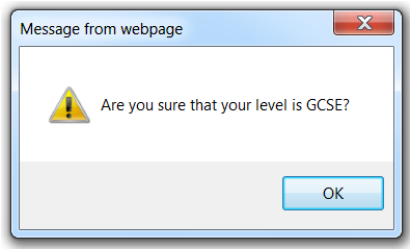
(w3schools, Window alert() method) (10)

</script> ends the JavaScript for the form and ends the validation.

These alert boxes show the alert message for all fields empty, the examinaton number is not exactly 4 digits long and they conformation question whether the candidate is sure with their choice in exam level (e.g GCSE)







DESCRIBE HOW THE HTML CALLS THE VALIDATION ROUTINE

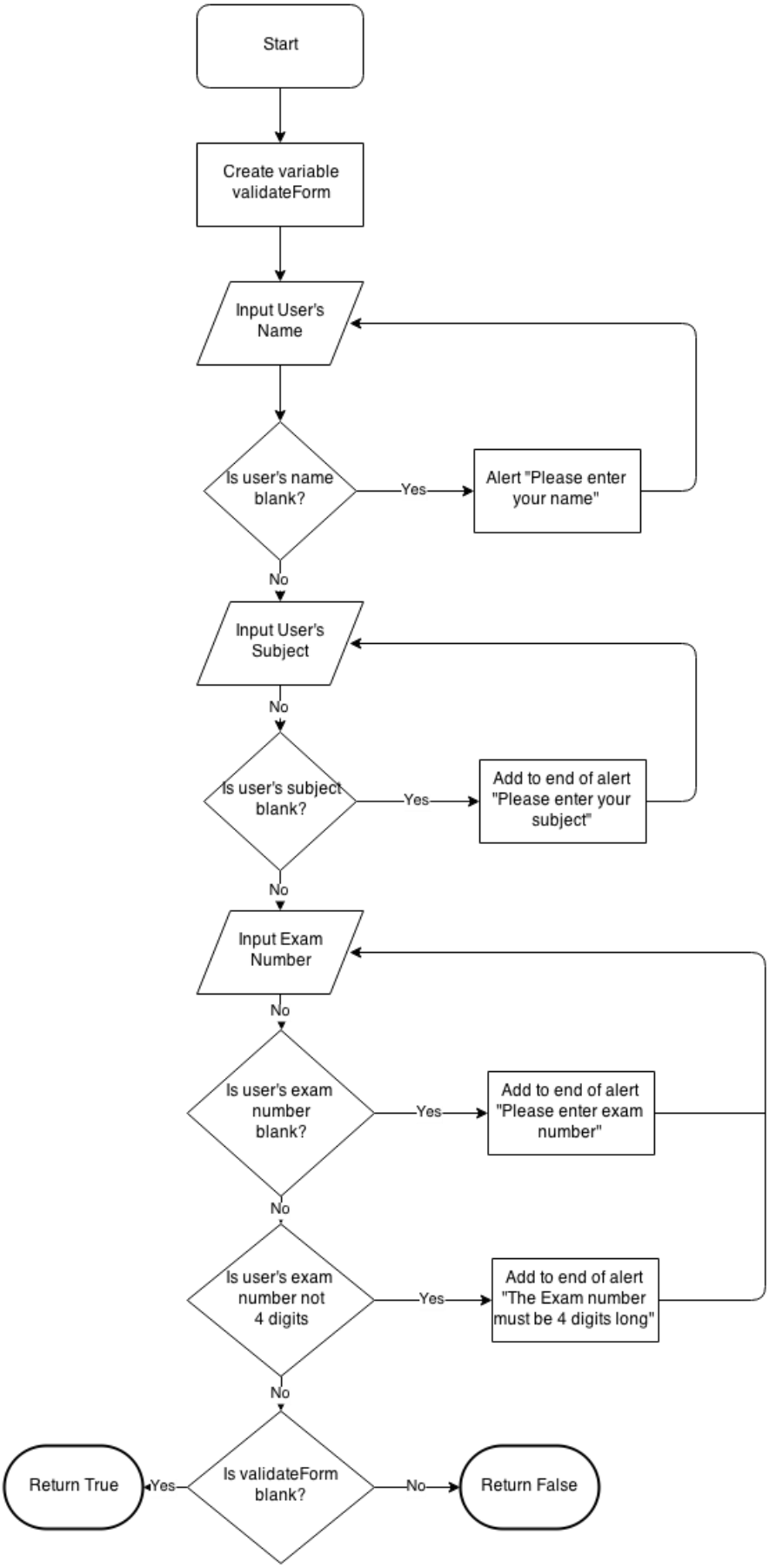
The code <td><input type="submit" name="Submit" value="Submit" onclick="return validateForm();" /></td> calls the validation routine. HTML doesn't call the validation routine. It is triggered by an event and this is done by javascript. When the user clicks the button 'Submit' on the form, the form data is sent to a JavaScript validation function 'validateForm' that checks each aspect of the input to make sure that it is in the appropriate format. This function is triggered by the onclick attribute. Each form element is evaluated according to criteria in the code. If the function finds an error in one of the fields, it should send a warning back explaining how the field doesn't fit in to the criteria in the code. (W3schools, Javascript Validation) (11)

CRITERIA FOR QUESTION 4

Test References	Reason for test	Test Instruction	Expected Result
All left blank	To test if alert message works	Leave all textbox's empty	Alert message appearing telling the user to fill in their name, subject and exam number.
Only name filled in	To test if alert message for subject and exam number work	Only enter name and leave other field empty.	Alert message appearing telling the user to fill in their subject and exam number.
Only subject entered	To test if exam number and name alert messages works.	Only enter subject, leave other fields empty	Alert message appearing telling the user to fill in their name and exam number.
Only name field empty	To test if name alert message works Independently	Enter subject and exam number, leaving name empty.	Alert message appears telling the user to enter their name.
Only subject field left empty	To test if subject alert message works independently.	Enter name and exam number and leave subject field empty.	Alert message asking the user to enter their subject.
Only exam number field left empty	To test if exam number alert message works.	Enter subject and name, leave exam number empty	Alert message asking to enter exam number.
Only exam number entered	To test if name and subject alert box work without exam number alert box.	Only enter exam number, leave other fields clear.	Alert message asking the user to enter name and subject.
All entered	To test if the action success page confirms success.	Enter your name, subject and 4 digits exam number.	Success page opened
Exam number 3 digits	To test if exam number if 4 digits alert box occurs.	Enter your exam number with 3 digits only	Alert box asking to enter 4 digits exam number.
Enter Exam number 5 digits	To test if exam number if 4 digits alert box occurs.	Enter your exam number with 5 digits only	Alert box asking to enter 4 digits exam number.

Enter Exam Number 4 digits long	To test if 4 digits alert box doesn't appear.	Enter you examination number with 4 digits	No alert box appears asking

QUESTION 4 FLOWCHART



QUESTION 4 PSEUDOCODE

```
Function validateForm
[
Var Alert
If document.examentry.name.value = “”
(Alert+=”You must enter your forename”)
If document.examentry.subject.value =””
(Alert+=”You must enter your subject”)
If document.examentry.examnumber.value=””
(Alert+=”You must enter your examnumber”
If document.examentry.examnumber.length=!4
(Alert+=”Your examnumber must be 4 digits long”
If Alert = “”
Return True
Else
Return false
]
```

ADD ANOTHER TEXT FIELD TO THE FORM TO TAKE THE USER'S EXAMINATION NUMBER

For this you must enter new fields were the user can enter their examination number. This was easy as the code was incredibly similar to the text fields for the name and the subject for the user.

This code sets the id to "Examination Number"

```
60.       <tr>
61.         <td id="examinationno">ExamNumber</td>
62.         <td> <input name="examinationno" length="4"
63.  type="text"></td>
64.       </tr>
```

This creates the textbox

This code will be displayed as the name the textbox

To make this code I used the original code (Figure 1) given to me to set the field for name and subject and modified the code to set the field as the position where the user enters their examination number. I changed the id to 'examnum' and changed the name next to the text field to 'ExamNumber'.

```
52.       <tr>
53.         <td id="name">Name</td>
54.         <td><input name="name" type="text"></td>
55.       </tr>
56.       <tr>
```

Figure 1

EXTEND THE JAVASCRIPT CODE TO VALIDATE THE FIELD TO MAKE SURE IT'S NOT BLANK

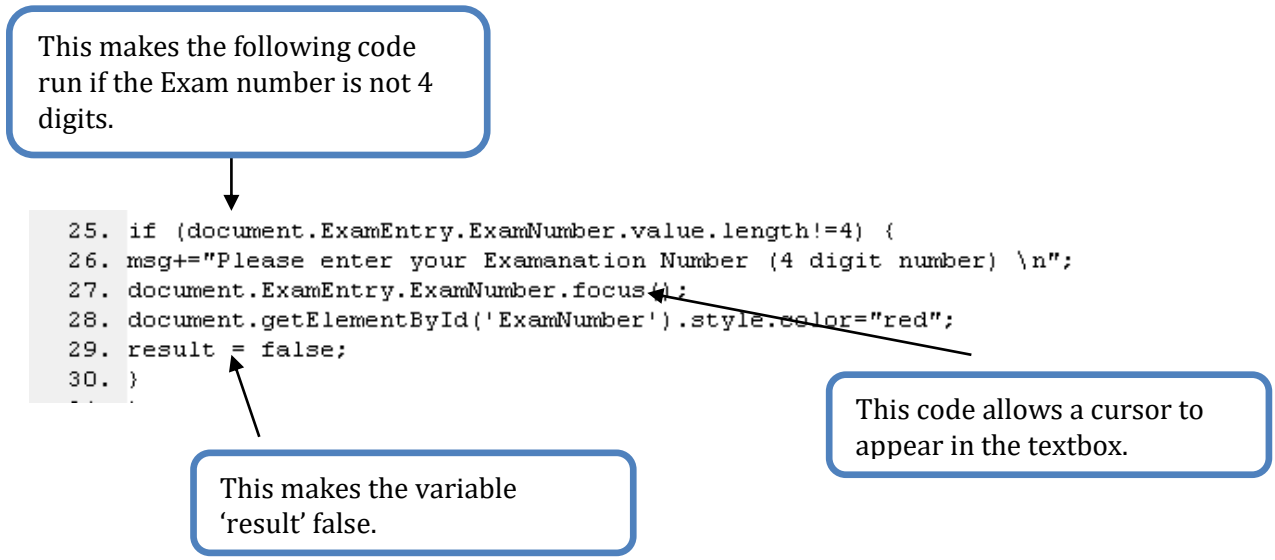
This runs if the exam number is blank.

```
20 if (document.ExamEntry.examnumber.value=="") {
21 msg+="You must enter your Examination Number \n";
22 document.ExamEntry.examnumber.focus();
23 document.getElementById('examnumber').style.color="red"
24 result=false;
25 }
```

This stops the page from carrying out the process.

This is the alert message that will show up.

EXTEND THE JAVASCRIPT CODE TO MAKE THE USERS EXAM NUMBER TO 4 DIGITS



QUESTION 4 TESTING

Test References	Reason for test	Test Instruction	Expected Result	Actual Result	Corrections
All left blank	To test if alert message works	Leave all textbox's empty	Alert message appearing telling the user to fill in their name, subject and exam number.	As expected (Figure 5)	none
Only name filled in	To test if alert message for subject and exam number work	Only enter name and leave other field empty.	Alert message appearing telling the user to fill in their subject and exam number.	As expected (Figure 6)	none
Only subject entered	To test if exam number and name alert messages works.	Only enter subject, leave other fields empty	Alert message appearing telling the user to fill in their name and exam number.	As expected (Figure 7)	none
Only name field empty	To test if name alert message works Independently	Enter subject and exam number, leaving name empty.	Alert message appears telling the user to enter their name.	As expected (Figure 8)	None
Only subject field left empty	To test if subject alert message works independently.	Enter name and exam number and leave subject field empty.	Alert message asking the user to enter their subject.	As expected (Figure 9)	none
Only exam number field left empty	To test if exam number alert message works.	Enter subject and name, leave exam number empty	Alert message asking to enter exam number.	As expected (Figure 10)	none
Only exam number entered	To test if name and subject alert box work without exam number alert	Only enter exam number, leave other fields clear.	Alert message asking the user to enter name and	As expected (Figure 11)	None

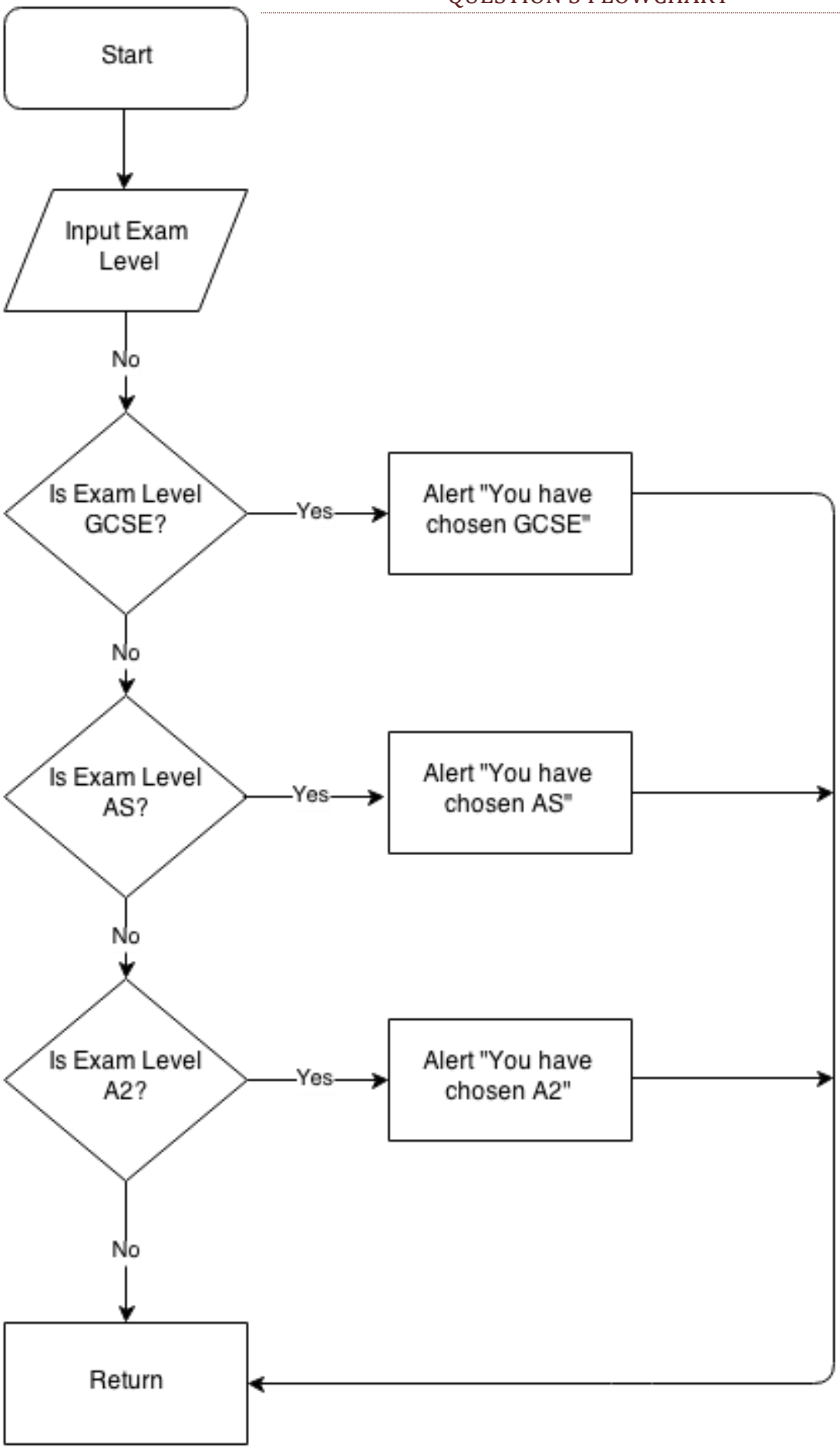
	box.		subject.		
All entered	To test if the action success page confirms success.	Enter your name, subject and 4 digits exam number.	Success page opened	As expected (Figure 12)	none
Exam number 3 digits	To test if exam number if 4 digits alert box occurs.	Enter your exam number with 3 digits only	Alert box asking to enter 4 digits exam number.	As expected (Figure 13)	none
Enter Exam number 5 digits	To test if exam number if 4 digits alert box occurs.	Enter your exam number with 5 digits only	Alert box asking to enter 4 digits exam number.	As expected (Figure 14)	none
Enter Exam Number 4 digits long	To test if 4 digits alert box doesn't appear.	Enter you examination number with 4 digits	No alert box appears asking	As expected	none

From this testing I can provide a conclusion that this code is fully efficient and effective as none of the tests failed.

ADD A SET OF RADIO BUTTONS TO ACCEPT A LEVEL OF ENTRY SUCH AS GCSE, AS OR A2.

Test References	Reason for test	Test Instruction	Expected Result
All left blank	To test if alert message works	Leave all textbox's empty	Alert message appearing telling the user to fill in their name, subject and exam number.
Only name filled in	To test if alert message for subject and exam number work	Only enter name and leave other field empty.	Alert message appearing telling the user to fill in their subject and exam number.
Only subject entered	To test if exam number and name alert messages works.	Only enter subject, leave other fields empty	Alert message appearing telling the user to fill in their name and exam number.
Only name field empty	To test if name alert message works Independently	Enter subject and exam number, leaving name empty.	Alert message appears telling the user to enter their name.
Only subject field left empty	To test if subject alert message works independently.	Enter name and exam number and leave subject field empty.	Alert message asking the user to enter their subject.
Only exam number field left empty	To test if exam number alert message works.	Enter subject and name, leave exam number empty	Alert message asking to enter exam number.
Only exam number entered	To test if name and subject alert box work without exam number alert box.	Only enter exam number, leave other fields clear.	Alert message asking the user to enter name and subject.
All entered	To test if the action success page confirms success.	Enter your name, subject and 4 digits exam number.	Success page opened
Exam number 3 digits	To test if exam number if 4 digits alert box occurs.	Enter your exam number with 3 digits only	Alert box asking to enter 4 digits exam number.
Enter Exam number 5 digits	To test if exam number if 4 digits alert box occurs.	Enter your exam number with 5 digits only	Alert box asking to enter 4 digits exam number.
Enter Exam Number 4 digits long	To test if 4 digits alert box doesn't appear.	Enter you examination number with 4 digits	No alert box appears asking
Radio button Alert Message	To test if an alert box appears when a radio button is clicked on.	Click on any of the 3 radio buttons.	An alert message asking if the was sure with your Exam Level
Only 1 Radio button can be clicked	To test if only one radio button can be clicked at one time	Click the first radio button the clicked the second and third.	The final radio button will only be checked.

QUESTION 5 FLOWCHART



QUESTION 5 PSEUDOCODE

```
Function alertMessageGCSE
[
If document.examentry.GCSE.value = “checked”
(Alert+=”You have selected GCSE as your level of entry, are you sure?”)
Return True
Else
Return False
]
Function alertMessageAS
[
If document.examentry.AS.value = “checked”
(Alert+=”You have selected AS as your level of entry, are you sure?”)
Return True
Else
Return False
]
Function alertMessageA2
[
If documentexamentry.A2.value =”checked”
(Alert+=”You have selected A2 as your level of entry are you sure?”)
Return True
Else
Return False
]
```

RADIO BUTTONS

```
<tr>
<td><input name="Level " checked="checked" value="GCSE" type="radio">GCSE</td>
<td><input name="Level" value="AS" type="radio">AS</td>
<td><input name="Level" value="A2" type="radio">A2</td>
</tr>
```

This code sets the GCSE radio button to be checked by default when the form opens.

The code name= sets the name of the radio buttons to "level"

This code sets the type to a radio button so it will appear on the form as a radio button.

This code sets the name next to the radio button to tell the user what the radio button is. (Figure 2)

Figure 2

I made the buttons through using w3schools radio button "try it" (Figure 3) and edited the code to fit the criteria for question 5(Figure 4). I had to change the name and the value; however I didn't have to edit the type as I wanted it the Levels to stay as radio buttons.

(w3schools, Forms and Inputs) (12)

```
<html>
<body>

<form action="">
<input type="radio" name="sex" value="male">Male<br>
<input type="radio" name="sex" value="female">Female
</form>

<p><b>Note:</b> When a user clicks on a radio-button, it
becomes checked, and all other radio-buttons with equal
name become unchecked.</p>

</body>
</html>
```

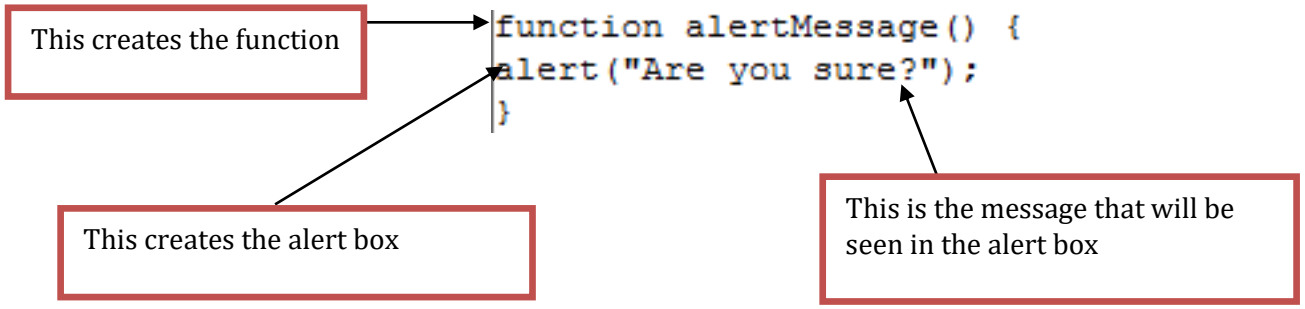
Figure 3

```
<!DOCTYPE html>
<html>
<body>

<form action="">
<input type="radio" name="level" value="AS">AS<br>
<input type="radio" name="level" value="A2">A2<br>
<input type="radio" name="level" value="GCSE">GCSE</form>
<p><b>Note:</b> When a user clicks on a radio-button, it
becomes checked, and all other radio-buttons with equal
name become unchecked.</p>

</body>
</html>
```

Figure 4

WRITE A FUNCTION THAT DISPLAYS THE LEVEL OF ENTRY TO THE USER IN AN ALERT BOX

I found the information to make an alert box from "w3schools" and I edited the alert message and name of the function to fit my code. When the user clicks on a radio button the alert box will appear asking the user if they are sure with their decision on their level of entry. The reason the alert box appears on click of the radio button is due to the code "onclick="return alertMessage". This means when the radio button is clicked the function alertMessage will begin and return. When JavaScript reaches a return statement, the function will stop executing. (w3schools).



Produce an evaluation of your solutions Q5 Testing

Test References	Reason for test	Test Instruction	Expected Result	Actual Result	Corrections
All left blank	To test if alert message works	Leave all textbox's empty	Alert message appearing telling the user to fill in their name, subject and exam number.	As expected (Figure 5)	none
Only name filled in	To test if alert message for subject and exam number work	Only enter name and leave other field empty.	Alert message appearing telling the user to fill in their subject and exam number.	As expected (Figure 6)	none
Only subject entered	To test if exam number and name alert messages works.	Only enter subject, leave other fields empty	Alert message appearing telling the user to fill in their name and exam number.	As expected (Figure 7)	none

Only name field empty	To test if name alert message works Independently	Enter subject and exam number, leaving name empty.	Alert message appears telling the user to enter their name.	As expected (Figure 8)	None
Only subject field left empty	To test if subject alert message works independently.	Enter name and exam number and leave subject field empty.	Alert message asking the user to enter their subject.	As expected (Figure 9)	none
Only exam number field left empty	To test if exam number alert message works.	Enter subject and name, leave exam number empty	Alert message asking to enter exam number.	As expected (Figure 10)	none
Only exam number entered	To test if name and subject alert box work without exam number alert box.	Only enter exam number, leave other fields clear.	Alert message asking the user to enter name and subject.	As expected (Figure 11)	None
All entered	To test if the action success page confirms success.	Enter your name, subject and 4 digits exam number.	Success page opened	As expected (Figure 12)	none
Exam number 3 digits	To test if exam number if 4 digits alert box occurs.	Enter your exam number with 3 digits only	Alert box asking to enter 4 digits exam number.	As expected (Figure 13)	none
Enter Exam number 5 digits	To test if exam number if 4 digits alert box occurs.	Enter your exam number with 5 digits only	Alert box asking to enter 4 digits exam number.	As expected (Figure 14)	none
Enter Exam Number 4 digits long	To test if 4 digits alert box doesn't appear.	Enter you examination number with 4 digits	No alert box appears asking	As expected	none
Radio button Alert Message	To test if an alert box appears when a radio button is clicked on.	Click on any of the 3 radio buttons.	An alert message asking if the was sure with your Exam Level	As Expected (Figure 15)	none
Only 1 Radio button can be clicked	To test if only one radio button can be clicked at one time	Click the first radio button the clicked the second and third.	The final radio button will only be checked.	All three checked- not as expected	Give all 3 radio buttons the same name so only one can be checked at one time.

This is helpful as it helps me conclude that my code is efficiently working and that the client-side JavaScript is effectively working by validating the empty fields and sending alerts. I has to repeat tests to make sure the code from the original testing still works

TEST PLAN FIGURES

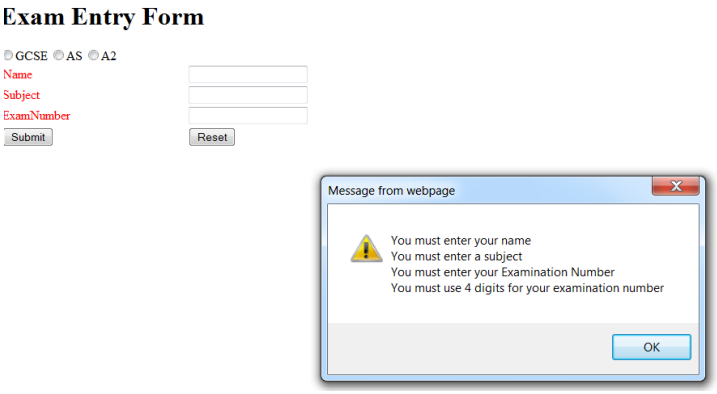


Figure 5

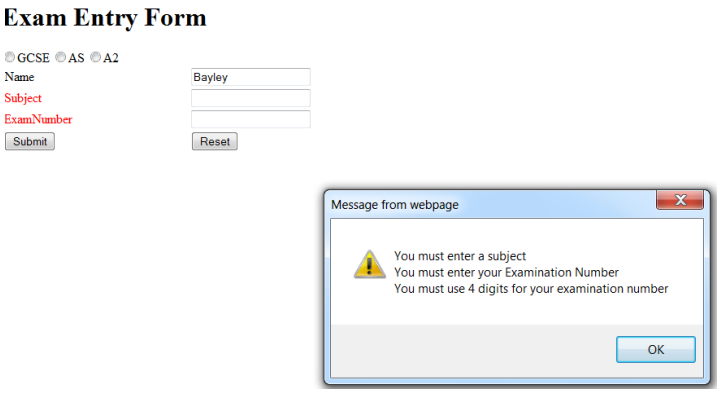


Figure 6

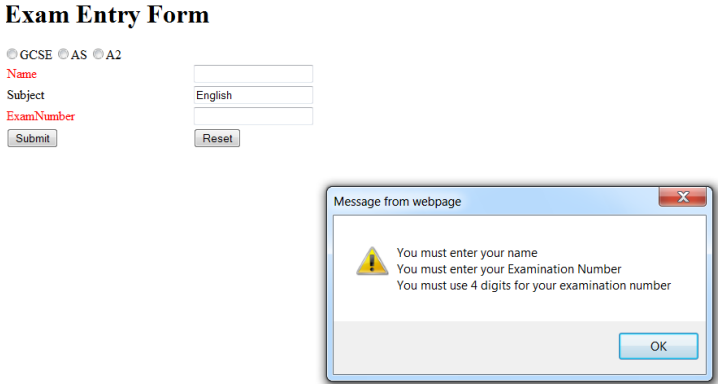


Figure 7

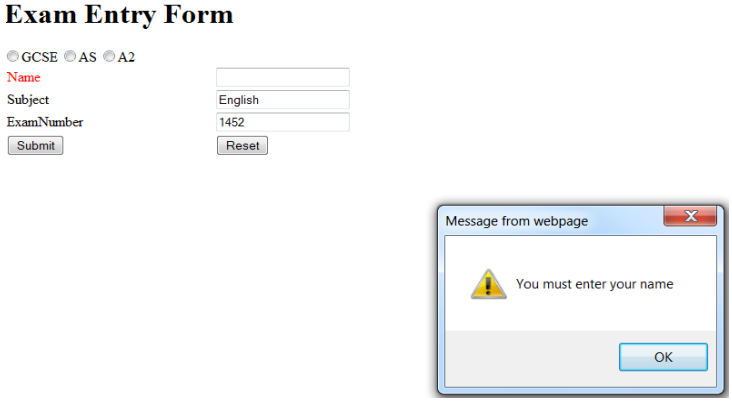


Figure 8

Exam Entry Form

GCSE AS A2
Name Bayley
Subject
ExamNumber 1452
Submit Reset

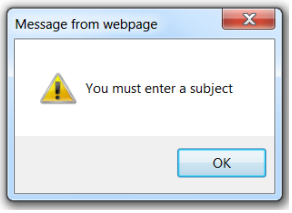


Figure 9

Exam Entry Form

GCSE AS A2
Name Bayley
Subject English
ExamNumber
Submit Reset

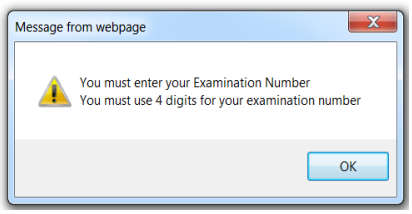


Figure 10

Exam Entry Form

GCSE AS A2
Name
Subject
ExamNumber 1452
Submit Reset

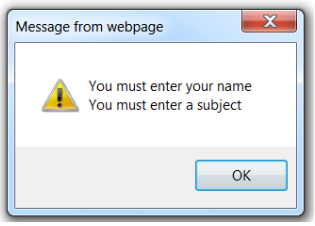


Figure 11

You entered all the data required

Figure 12

Exam Entry Form

GCSE AS A2
Name Bayley
Subject English
ExamNumber 145
Submit Reset

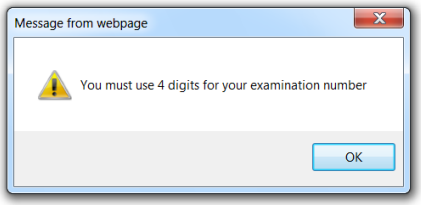


Figure 13

Exam Entry Form

GCSE AS A2
Name Bayley
Subject English
ExamNumber 14523
Submit Reset

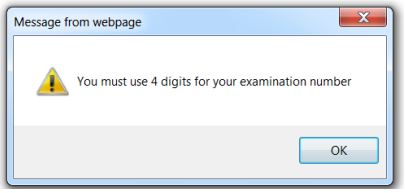


Figure 14

Exam Entry Form

GCSE AS A2
Name Bayley
Subject English
ExamNumber 1453
Submit Reset

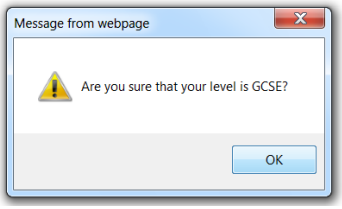


Figure 15

ALERTING THE USER ABOUT THEIR LEVEL OF ENTRY

To alert the user about if they are sure about picking their level of entry I created three separate functions for each level of entry. I called each function alertMessage and added the level of entry at the end of the function name so the name is suitable for each function e.g. for GCSE I created the function alertMessageGCSE. Each function only had to add an alert to each time the user clicks a radio button all I had to was add the onclick on the end of the radio buttons so when the user clicks, on the click the function configures an alert to ask the user if they are sure with their decision.

PRODUCE AN EVALUATION FOR YOUR SOLUTIONS

Q4) For Question 4 part I, I looked at the previous code given to me, and edited it to fit what the question asked for. So I duplicated the code (Figure 16) and edited the id, tag and name so the new field for text will now take the user's examination number, rather than their name which made (Figure 17). To create this examination number field, I did not edit the type as by changing "text" as it would not create a textbox/field. The <tr> define a row in a table so must be kept so to keep the alignment of the text boxes, so I must not edit/delete them so the text boxes stay together.

In conclusion, by using previous code to create a text field, I created a new field for the examination number of the candidate so they can be verified if they are doing the correct course.

```
<tr>
  <td id="examnum">ExamNumber</td>
  <td> <input name="examnum" type="text"></td>
</tr>
```

Figure 16

```
<tr>
  <td id="name">Name</td>
  <td><input name="name" type="text"></td>
</tr>
```

Figure 17

Q4) For Question 4 part II, again, I looked at previous code (Figure 18) and found that my solution was very similar to code I was given. To validate the field/textbox I looked at the code that validated the field for forename and edited parts of the code such as the id from "name" to" examnum" and by changing every "name" to "examnum" (Figure 19). I also had to edit the alert since the user will understand that they must enter their exam number for the form to return true to the success page to tell the user that they have completed the form with no errors. I must not change parts of the code such as "return = false" as if the user has not enter their name/examination number/ subject, without the "return = false" the form will send it off as true to the "success.html" page which should not happen as they have made an error in the form and must be alerted that they must enter data into every field with the correct characters. The ease of use is very high as people can easily enter their exam number in the correct field as it is labeled Examination Number. They will also be able to understand what they need to do as the alert box is very clear on the instructions they must forfill in order to send the form off.

In conclusion I used previous code given and slightly edited to produce an efficent and functional solution to the task.

```
if (document.ExamEntry.name.value=="") {
msg+="You must enter your name \n";
document.ExamEntry.name.focus();
document.getElementById('name').style.color="red"
result = false;
```

Figure 18

```
if (document.ExamEntry.examnum.value=="") {
msg+="You must enter your Examination Number \n";
document.ExamEntry.examnum.focus();
document.getElementById('examnum').style.color="red"
result=false;
```

Figure 19

Q4) For Question 4 part III I looked at the code I had recently just made (Figure 20) and edited it so it would fit the requirements of the question. Since the question asks for the length to be validated, since we have validated that the field must not be left blank, this will only talk about entering 4 digits. By duplicating the code for Question 4 II and editing the alert to tell the user the examination number must be 4 digits long, now the user will know that the length of their examination number is the error. Since we are not looking for the value of the field, we are looking for the length so we add ".length" on the end of the IF statement so it will apply to the length of the digits in the field. By going on w3schools I found that by using the "!" it states that it must be exactly that digit, so by putting "!=4"this means the examination number must be exactly four digits long which fits with the requirement. To help me test this I used w3schhols try it section so I could easily edit and adapt the code to fit perfectly with the specification. This created (Figure 9).

```
if (document.ExamEntry.examnum.value.length !=4) {
msg+="You must use 4 digits \n";
document.ExamEntry.examnum.focus();
document.getElementById('examnum').style.color="red"
result = false;
}
```

Figure 20

In conclusion through looking through websites like w3schools helped me understand how you can validate a field so it must take the exact amount of the number.

Q5) for this task, I went to w3schools to find out how to create a radio button. Here I was able to create three radio buttons, one of each level and giving each level the same name "Level" so only one radio button could be pressed at one time and this is why in my question 5 completed plan, my final test did not go as expected as I gave each radio button a different name, so they could all be checked at the same time. To give the radio button its own value, I changed the value of each radio button to the level they represent e.g. GCSE so the form could distinguish them (Figure 21). By creating 3 separate functions, alertMessageGCSE, alertMessageAS and alertMessageA2 it distinguishes the three separate alert messages that will send to the user if the click radio button GCSE, AS and A2. By having a separate message per Level, it will identify to the user if they are sure with the level they have chosen e.g. if they have chosen GCSE the alert will mention that they have chosen GCSE and will ask the user if this is the correct level for their subject. I used a function of the alerting because when JavaScript reaches a return statement, the function will stop executing. If the function was invoked from a statement, JavaScript will "return" to execute the code after the statement. Functions often create a return value. The return value is "returned" back to the original form which can also be known as the 'caller' as it called the function.

Overall through simply researching why to use functions to ask for the user's conformation if they were sure with the decision they made on choosing their level of entry. Through using a function it made the code more efficent as only two lines of code were needed for each Radio button. I could of made it more efficent by only using one function with the same alert message "are you sure", however I belive that it did not give a clear enough explanation to the user on what they had to do to, so the function would return the result as true, so it continued to the sucesss page.

Figure 21

Question	What the question asked for	What I did	Code
Four Part I	Question 4 part I asked for an extra field so the user could enter his examination number.	I looked at the previous code to create the field for the name and subject of the user and edited it slightly so the id and name will be for 'Examination Number' As the field is for text, the 'type' will not be changed.	<tr> <td id="examnum">ExamNumber</td> <td> <input name="examnum" type="text"></td> </tr>

Four Part ii Four Part iii	The question asked for examination number field to validate if the field is blank and to alert the user if the field is blank and to tell the user, to continue, the user must enter their examination number	I looked at previous code for validating the name field and edited parts of the code so it will work for the examination number field. The parts of the code I had to edit were the id of the examination number text field so it will know which text field to validate whether it is blank or not. I also had to edit the alert so it will be able to tell the user what exact text field cannot be left blank.	if (document.ExamEntry.examnum.value==="") {
			
			msg+="You must enter your Examination Number \n";
			
			document.ExamEntry.examnum.focus();
			
			document.getElementById('examnum').style.color="red"
			
			result=false;

Four Part iii	The question asked to validate the examination number field to only accept 4 digits for the examination number to be valid.	I went on w3schools and saw that to make the code only allow an exact number of digits you must use the "!" which means not exactly. I used some of the same code for Question Four part 2 but on the first line of code, I had to add ".length" as it asked for the length to be 4 digits not the value. I had to edit the alert so the user can know that the examination must be exactly 4 digits not any other number.	if (document.ExamEntry.examnum.value.length !=4) {
			
			msg+="You must use 4 digits \n";
			
			document.ExamEntry.examnum.focus();
			
			document.getElementById('examnum').style.color="red"
			
			result = false;

Five	This question asked for me	I went to w3schhols	<input name="Level" level="1"

	to create 3 radio buttons one for each level of entry and to alert the user if they are sure with their choice of level.	and found out how to make radio buttons. Originally it allowed all three radio buttons to be pressed so I had to change the name to be the same so I could only check one radio button at a time so only one level can be checked. To send a message asking if the user is sure with their choice of level of entry, I used 3 separate functions, one for each level, which gives a specific alert whether the user is sure in choosing the correct level of entry they are participating in.	value="GCSE" onclick="return alertMessageGCSE();" type="radio">GCSE
			
			<input name="Level" level="2" value="AS" onclick="return alertMessageAS();" type="radio">AS
			
			<input name="Level" level="3" value="A2" onclick="return alertMessageA2();" type="radio">A2
			
			function alertMessageGCSE() {
			alert("Are you sure that your level is GCSE?");
			}
			function alertMessageAS() {
			alert("Are you sure that your level is AS?");
			}
			function alertMessageA2() {
			alert("Are you sure that your level is A2?");
			}

FINAL CODE

```
//Start of the form
<html><head>
//The title of the form will be Exam Entry
<title>Exam Entry</title>
//The script language will be JavaScript so at the end of the script, the JavaScript will stop
 <script language="javascript" type="text/javascript">
//Create the function to alert the user of their certainty of GCSE, and if it's their correct level
function alertMessageGCSE() {
alert("Are you sure that your level is GCSE?");
}
//Create the function to alert the user of their certainty of AS, and if it's their correct level
function alertMessageAS() {
alert("Are you sure that your level is AS?");
}
//Create the function to alert the user of their certainty of A2, and if it's their correct level
function alertMessageA2() {
alert("Are you sure that your level is A2?");
}
//Create the function validateForm so the empty fields can be validated, if there are no errors
the form will continue with its action and the result of continuing will be true
function validateForm() {
var result = true;
var msg="";
```

```
//If the user's name field is empty, alert the user to enter their name
if (document.ExamEntry.name.value=="") {
msg+="You must enter your name \n";
document.ExamEntry.name.focus();
document.getElementById('name').style.color="red"
result = false;
}
//If the subject field is empty alert the user to make sure the user adds their subject to continue
if (document.ExamEntry.subject.value=="") {
msg+="You must enter a subject \n";
document.ExamEntry.subject.focus();
document.getElementById('subject').style.color="red";
result = false;
}
//If the exam number text field is empty, alert the user to make sure there is data in the field
if (document.ExamEntry.examnum.value=="") {
msg+="You must enter your Examination Number \n";
document.ExamEntry.examnum.focus();
document.getElementById('examnum').style.color="red"
result=false;
}
//If the exam number is not exactly four digits, alert the user telling them to make it 4 digits
if (document.ExamEntry.examnum.value.length !=4) {
msg+="You must use 4 digits for your examination number \n";
document.ExamEntry.examnum.focus();
document.getElementById('examnum').style.color="red"
result = false;
}
```

```
//This code tells the JavaScript is the fields are empty to continue and send the alert message set in the If statement.

// It will only send the alert message if that field is empty

if(msg==""){

return result;

}

{

alert(msg)

return result;

}

}


//This ends the JavaScript part of the Form

</script>

</head>

//The start of the HTML

<body>

//This creates the heading 'Exam Entry'

<h1>Exam Entry Form</h1>

//This creates the Radio Buttons for the levels GCSE, AS and A2 with the function to alert the user if they are sure with their decision on click of the radio button

<input name="Level" level="1" value="GCSE" onclick="return alertMessageGCSE();" type="radio">GCSE

<input name="Level" level="2" value="AS" onclick="return alertMessageAS();" type="radio">AS

<input name="Level" level="3" value="A2" onclick="return alertMessageA2();" type="radio">A2

//This names the form and the action that will occur after the page has been validated and has passed the validation stage.

<form name="ExamEntry" method="post" action="success.html">
```

```
//This creates an invisible table (as the border is 0) which the text fields and labels will be inside to keep a neat alignment

 <table 0="" border="0" width="50%">

   <tbody>

    <tr>

//This creates the name field with a Label next to it which says 'name' so the user knows to enter their name in the corresponding text field.

 <td id="name">Name</td>

      <td><input name="name" type="text"></td>

    </tr>

    <tr>

//This creates the subject field with a Label next to it which says 'subject' so the user knows to enter their subject in the corresponding text field.

 <td id="subject">Subject</td>

      <td> <input name="subject" type="text"></td>

    </tr>

    <tr>

//This creates a text field for the Examination Number and a Label next to it called 'Exam Number' so the user knows to enter their Examination Number into this field.

<td id="examnum">ExamNumber</td>

      <td> <input name="examnum" type="text"></td>

    </tr>

    <tr>

//This creates the submit button and the reset button to submit the final version of the form and to validate the form on click of the button.

//The reset is set to reset the whole form if the user makes a mistake or several

<td><input name="Submit" value="Submit" onclick="return validateForm();"
type="submit"></td>

      <td><input name="Reset" value="Reset" type="reset"></td>

    </tr>

   </tbody>

 </table>
```

```
</form>
</body>
//this ends the page as the HTML is being ended
</html>
```

WRITE A CONCLUSION ABOUT THE EFFECTIVENESS OF VALIDATION ROUNTINES TO REDUCE THE NUMBER OF ERRORS THAT ARE MADE IN DATA INPUT

(13)

What is JavaScript?

I thought that it would be good to mention what JavaScript is so it gives me a good understanding about the script language and how the how it can differ a web pages content.

"A scripting language developed by Netscape to enable Web authors to design interactive sites. Although it shares many of the features and structures of the full Java language, it was developed independently. JavaScript can interact with HTML source code, enabling Web authors to spice up their sites with dynamic content." (/13)

This gives an understanding of what JavaScript is and its primary role.

(14)

To be able to talk about the effectiveness of the JavaScript, I believe that by adding the advantages of using JavaScript over another script language such as PHP as well as the disadvantages of using this un-secure script language.

Advantages of JavaScript

"Being client-side the speed of JavaScript is very fast because any code functions can be run immediately instead of having to contact the server and wait for an answer.

JavaScript is very simplistic as it is simple to learn as well as being able to implement it.

Since JavaScript is client-side the server load reduces the demand on the website server.

JavaScript plays nicely with other languages and can be used in a huge variety of applications. Unlike PHP, JavaScript can be inserted into any web page regardless of the file extension. JavaScript can also be used inside scripts written in other languages such as PHP.

Disadvantages of JavaScript

JavaScript has a low security level since the code executes on the users' computer, in some cases it can be exploited for malicious purposes. This is one reason some people choose to disable JavaScript.

JavaScript has a low level of reliability as JavaScript is sometimes interpreted differently by different browsers. Whereas server-side scripts will always produce the same output, client-side scripts can be a little unpredictable." (/14)

This effectively conveys that Client-Side JavaScript can execute its job effectively at checking if fields are empty or if any data inputted is not acceptable.

(Media College, 2014)

(15)

As most validation of JavaScript is Client-Side, this gives a deeper understanding of the script language and how it can validate a web page's empty fields effectively.

"JavaScript is executed on the client side. This means that the code is executed on the user's processor instead of the web server, saving bandwidth and strain on the web server.

The JavaScript language is relatively easy to learn as it comprises of syntax that is close to the English language. It uses the DOM (document object model) that provides plenty of prewritten functionality to the various objects on pages making it a breeze to develop a script to solve a custom purpose." (/15)

In conclusion JavaScript executed on client-side is effective at validating the user's errors and also saves bandwidth as it does not have to send the information to the server. However due to the validation being client-side no information like house number and street name can be validated to be real, but if the field has data in it.

(JS Scripters, 2014)

(16)

This gives a more technical approach to how JavaScript can be used effectively and the pros and cons that come with the popular script language

Advantages with JavaScript

"Less server interaction: You can validate user input before sending the page off to the server. This saves server traffic, which means fewer loads on your server.

Immediate feedback to the visitors: They don't have to wait for a page reload to see if they have forgotten to enter something.

Increased interactivity: You can create interfaces that react when the user hovers over them with a mouse or activates them via the keyboard.

Richer interfaces: You can use JavaScript to include such items as drag-and-drop components and sliders to give a Rich Interface to your site visitors."

The limitations with JavaScript

"Client-side JavaScript does not allow the reading or writing of files since this is kept for security reason.

JavaScript cannot be used for networking applications because there is no such support available.

JavaScript doesn't have any multithreading or multiprocessing capabilities. "(/16)

I can conclude that even though JavaScript has limitations of low security, it can effectively validate a webpage for errors made by the user that are seen as unacceptable or against the criteria.

(Tutorials Point, 2014)

(17)

This vividly portrays the reasons to use JavaScript in a Web Page and for other examples such as Dashboards, Photoshop etc. However this also highlights the reasons not to use JavaScript and how it may not be seen as effective in validating empty fields in a Web Page.

Reasons to use JavaScript

Load and Go Delivery

"It doesn't make any .exe, or any abstract class files. So, it throws errors and exceptions only while executing that particular path of the program. This approach has two advantages.

First, it helps debugging. In case of languages like JavaScript, where you don't have to use any IDE, this approach actually helps you in proper line by line debugging.

Second, if some other part of the same program is under construction, then also you can run the other flows of the program.

JavaScript has introduced many special features in its functions which makes them different.

A function is able to accept any type or arguments and also can return any type of value. They can accept any number of arguments even without declaring. This flexibility reduces a lot of codes in a project.

JavaScript is also not only for web pages, it can also be used in:

- Dashboards
- Dreamweaver
- Photoshop
- PDF editors
- Articulate
- Yahoo Widgets

However JavaScript disadvantages are:

JavaScript only has single inheritance; multiple inheritances are not possible.

No copy or equal method is there in JavaScript.

The bit wise operators in JavaScript convert the operand in 32-bit signed integer and turn the result back to 64-bit floating point. So, bit wise operation is slower here.

JavaScript is different than other languages and according to me, a language should be different, otherwise there's no meaning of its existence. JavaScript gives you few extra ordinary features, whereas it also has some cons in it, but it's all up to the developer how smartly he uses a language.

Now, JavaScript is not alone. It has jQuery, data binding frameworks and many other tools in its family. Day by day, people are reducing the dependency in server side and trying to make a web app more dependent on client side. These approaches are the first sign of the bright future of JavaScript". (/17)

This concludes that JavaScript is an effective language that can validate the user's errors.

(Void Canvas, 2014)

By asking Mr Cowen, "In a web page, would use validation or verification to validate/verify the user's inputs to be valid"

" *Due to Validation being client-side it is much quicker and easier to use than Verification, as it does not have to be sent to the server, as Verification is server-side, so validation can immediately run functions rather than wait for the server to respond with the answer. However due to being server-side, the answer is more precise, so verification is more reliable than server-side validation. Overall, I would use Verification instead of Validation, due to it being more reliable and is much more precise so I know my data will have a higher chance of being valid."*

I believe this is helpful information to know as Mr Cowen is a professional web page developer and uses script languages such as JavaScript Server-side to validate web pages with empty fields. However he only uses JavaScript with Web Pages that contain information that wouldn't be helpful for a hacker such as Job Applications.

(Cowen, 2014)

(18)

I believe this is important as it highlights the effectiveness of server-side JavaScript validation and the efficient processes it would go through if the user's data is not valid as it has been validated by the server as that contains public information, that client-side validation does not contain.

"Server-side validation scripts do not rely on or fret with JavaScript support in the user's web browser; consequently, they are unaffected by the presence or absence of JavaScript support in the user's web browser. With server-side validation, information is sent to the server and validated using one of many server-side languages.

If the validation fails, the response message is then sent back to the client, and the page that contains the web form is refreshed and a feedback message is displayed. This is a more secure method and works even if JavaScript is turned off in the browser and it is not easily bypassed by users with malevolent intent.

The downside with server-side validation is that users will have to fill in the information without getting a response, only until they click the submit button, and this typically results in a slower overall response time for the user experience." (/18)

(Boudreaux, 2014)

Overall server-side validation is efficient and effective at validating if the user's inputs are valid, but due to the lack of security it holds it may not be a safe language to use for credit card numbers.

(19)

I decided to quote this source as it conveys the ways JavaScript is efficient and effective for validating if fields are empty or the user's data does not match with the server's data so calling the routine as false as the data is invalid. It clearly explains the limitations with JavaScript as it cannot recognize acceptable values without the value being entered into the criteria as unacceptable or acceptable.

"The JavaScript validation programming interface consists of four components:

- Low-level processing functions for numerical values, text, and whitespace.
- Validation functions for each type of data.
- Validation handler used for 'onsubmit', which supports generic form validation using a list of form elements, or fine-tuned validation by using a validation profile customized for your form.
- Definitions of error codes and user-friendly error code messages.

Data Validation precautions

A serious concern with data validation is that a validation script sometimes is unable to recognize acceptable values, or that it may be non-functional depending on browser settings or other issues. If an error occurred when loading a script in a page, a customer might not be able to submit the order. In addition, it tends to make security managers forget that malicious users can intentionally disable scripts. Nothing from a form should ever be blindly trusted if a critical area of security might be compromised if an unexpected value was submitted.

Available Validation Functions

"Each validation function sets an error code if there's an error, and it has to return an array of valid values (possibly with prettifications and reformatting) if there's no error. These validation functions were written (for the most part) before this validation script or article existed. They've been tested for years to make sure that the validation script is of production quality." – This allows the validation routine to be more effective and minimizes errors on the data input as the functions have a pre-set error code.

Function	
Cardno	Validation macro to define for a field for the zip values, either five digits or nine digits; as with other macros, it recognizes several variants, including spaces or no spaces, and with the case of a zip code, an optional hyphen. It is possible for multiple zip codes to be verified.
Email *	Validates all kinds of email addresses. It is possible for multiple email addresses to be validated.
Expires	checks a four-digit expiration date, that it has not expired, and that it is not an invalid amount into the future (using the computer's own clock, which is occasionally mis-set, resulting in errors, but this is OK, since the customers will be allowed to judge for themselves after receiving one warning).
Tel	Telephone numbers, including long distance and country codes, and all kinds of separators.
Text	Checks that text (this does not mean alphabetical characters only) is

	present.
URL	Matches the wide range of possible URL values. Of course, it is possible that a URL may point to a non-existent location. It may return multiple values.
Words	Checks that at least two words of text are present.
Zip	Validation macro to define for a field for the zip values, either five digits or nine digits; as with other macros, it recognizes several variants, including spaces or no spaces, and with the case of a zip code, an optional hyphen. It is possible for multiple zip codes to be verified.

*Note that the actual return values of all of them are arrays; it's possible for some to return more than one value in the array, such as with email() for email addresses. Also note that the return value might not be an array if there has been an error code set.

These Functions are only used on server-side validation as the data inputted must be sent to the server to validate whether the data is true and or valid e.g. To validate whether the zip code inputted is a valid zip code in the country or if it is made up. This validation must be server side as the data is not available to be validated client-side, so the validation has an effective routine and no errors." (/19) (Myers, 2014)

I can conclude that JavaScript is an effective language to use to validate if fields are empty or data inputted by the user is valid

(20)

I have quoted this as I believe it gives a good understanding of the disadvantages of JavaScript validation and the cost it may create by overusing it to finally become an inflexible resource to use in your form.

"Users can generate errors by entering invalid data, selecting invalid program options, or performing operations out of sequence. You can detect any of these error conditions by validating user input. However, in most cases you'll only want to use validation to detect data entry errors. Overusing validation can be expensive in terms of processing power and can lead to inflexibility in application design because it often ties application logic to the user interface.

For instance, suppose you were to perform some validation each time users select a program option, such as "Edit Employee Records," to make sure that they're allowed to select that option from the page they're on or from the point they're at in some sequence of operations. Validating every one of these operations for every user of the system would require a lot of conditional code and would also effectively hard-wire application sequences and interfaces to the underlying code. It would be far more effective to prevent user interaction errors than to detect them after the fact. From a user interface perspective, it's much more effective to design an interface that guides users through valid choices than to build one that inundates them with error messages every time they select an invalid option. Again, such decisions are fundamental to how you design your user interface and must be made during the design phase of a project.

For data entry errors, validation works well, and in many cases must be performed to prevent bad data from being sent to a critical system, such as a database. Data entry validation involves developing acceptance criteria that identify the type and format of data allowed for each data

field. Once the acceptance criteria for all data fields have been determined, you can write routines that check to see if data entered fits the acceptance criteria." (/20)

Here I have been able to show that even though the client-side effectiveness of JavaScript may give a quick and simple solution to validate empty fields it can be made inflexible by overusing. To overuse it you must set the criteria to a lengthy process which will make no difference in speed between Client and Server-Side as they will both be lengthy processes to validate each part of the criteria.

(Kuslich, 1999)

Validation and Verification

(21)
To give myself an idea of the main differences and similarities of Validation and Verification I used the OCR Edition GCSE Computing Textbook.

"Validation is the process where data is checked as it is entered into the system. This checking is done by the data handling software and it can be set up to reject any data that does not conform to specified rules.

These rules can be whatever the administrator wants them to be. Common examples such as digit checks or formatting checks can be useful when data is being copied in to a computer system by a human operator.

These are only common examples of validation checks. The software can be set up to check for many other potential errors. It is very important to realise that validation does not prevent all errors, it only detects rule breaking.

Validation Check	Explanation on how the validation check works
Check digit	A calculation is performed on a number that generates another digit which is added to a number. This is commonly seen on account numbers bar codes and the ISBN of a book. When the data is entered, the calculation is repeated and if the same check digit isn't generated, the input is rejected, however if the calculation is repeated and the same check digit is generated, the input will be accepted.
Format check	The data might have to confirm to a particular pattern. For example, the modern day UK car registration plate pattern is in the format LLNN LLL where L is a letter and N is a number.
Length check	The data must be within a certain limits. A telephone number might be setup to take no longer than twelve digits.
Lookup check	Some fields might be looked up to see if they are from an allowed range. Postcodes can be checked to see that they do in fact exist.
Presence check	This is to ensure that all important information has been included. An application for a bank account must have a surname, for example.
Type check	The data type must be restricted. For example, a person's surname might be restricted so that no numerical values are accepted and for the age of the person, no characters are accepted
Range check	A number must fall within limits. For example, for a popular event, each applicant might be limited to two tickets only.

Verification is a process that is used to ensure that the data sent is correct. In other words, data in the database is the same as in reality. Often, the only way to ensure this is through human inspection; however Humans make mistakes too easily. A slower, but more reliable method is to enter the same data twice, using two operators. The software can compare the two versions and highlight any differences.

It is more accurate and faster to enter data by machine, so there are numerous ways in which the data entry can be automated. Some examples are Bar Codes, Smart Cards, Magnetic Strips, OCR, OMR and RFID." (/21)

From this I have been able to explain that even though verification is a much slower process than Validation, it is more reliable and precise and could be used over JavaScript validation, however due to its lengthy process having to be sent to the server, it may not be able to match the quick, Client-Side JavaScript validation.

(Rouse & O'Byrne, 2012)

PHP vs Node.js

(22)

To determine if JavaScript is effective at validating forms, I had to compare it with other script languages that can be used to validate a form server and/or client-side.

“PHP was an outstanding technology in its days. Its success and popularity came from:

- Its ease of learning and use
- cheap and straightforward hosting mostly shared LAMP
- Abundance of open-source scripts, apps and libraries

At the same time, these same things now led to its dusk. The contributions to the core from beginner programmers metamorphosed API inconsistently while the lack of OOP/classes and module management systems inhibited open-source community growth. Absence of a leading framework (Ruby on Rails comes to mind as an example of a single dominance) or a paradigm that also helped to produce a lot of bad code that relied heavily on mixing PHP and HTML code without any MVC. On the other hand, there are a lot of good products and infrastructure for PHP that are here to stay.

Node.js is relatively young with only three years since its first commit, but it’s already the fastest growing platform by the pace of contributions (the absolute number will surpass other languages in a few years). The fact that JavaScript language is the most popular language in the world and has the biggest run-time internment of course attributed to that. Many tools are ported to Node.js with small or no modification from the browser environment. Also, great books on JavaScript fundamentals (for example, JavaScript: The Good Parts and Eloquent JavaScript) experienced surge in the popularity again.

Node.js is very efficient and great for building real time, NoSQL oriented and scalable systems”. (/22)

Overall I found out that Node.js is an effective way of validating a form compared with PHP, a script language which is slowly being over-taken as often choice to using a script language to validate a form.

(Azat, 2014)

Group Work

Mr. Fennel came into our school and answered a variety of questions about the advantages and disadvantages of JavaScript client and server-side.

“JavaScript is an effective way to validate a form, Client-side due to being able to find errors in the form, only if it is defined as an error. For security it is not the best, JavaScript has a low security level since the code executes on the users' computer, and can be exploited for harmful purposes. This is just one reason some people choose to disable JavaScript for this low security.”

From this I learnt that JavaScript is an effective and efficient script language for validating a form of a person’s name and subject for their school, but due to its low level of security you wouldn’t use it for the user’s credit card number.

CONCLUSION

I can conclude that there is little doubt that my system is fully functional and efficient. I believe that JavaScript is very effective at calling the validation routine ad minimizing errors, however it is very important to realize that validation does not prevent all errors, it only detects rule breaking. From the group work I learnt that due its low level of security it minimizes the use of JavaScript in industry as the data could be easily hacked and used for malicious purposes. From my conclusions, I can conclude that JavaScript is an efficient and an effective script language to reduce the number of errors the user inputs as it effectively can validate if fields are empty with Client-Side Validation or if his house number, post code or their registration plate is a valid input that exists in the server's database. As it has got a low-level of security, it should not be used to validate credit card numbers or passwords, but for information that hackers cannot do anything with. Also from my Test Plans, I have concluded that due to all but one test's result was as expected from the assumption on what it should do I can conclude that it is effective in identifying errors set by the user.

Bibliography

Number of Citation	URL/Book
1	http://www.w3schools.com/tags/tag_td.asp
2	Microsoft.com
3	http://www.w3schools.com/js/js_functions.asp
4	http://www.w3schools.com/js/js_variables.asp
5	http://www.w3schools.com/jsref/jsref_regexp_newline.asp
6	http://www.w3schools.com/jsref/met_html_focus.asp
7	http://www.w3schools.com/js/js_htmldom_html.asp
8	http://www.w3schools.com/jquery/event_focus.asp
9	http://www.w3schools.com/js/js_booleans.asp
10	http://www.w3schools.com/jsref/met_win_alert.asp
11	http://www.w3schools.com/js/js_form_validation.asp
12	http://www.w3schools.com/html/html_forms.asp
13	http://www.webopedia.com/TERM/J/JavaScript.html
14	http://www.mediacollege.com/internet/javascript/pros-cons.html
15	http://www.jscripters.com/javascript-advantages-and-disadvantages/
16	http://www.tutorialspoint.com/javascript/javascript_overview.htm
17	http://voidcanvas.com/reasons-to-use-javascript-with-pros-and-cons/
18	http://www.techrepublic.com/blog/web-designer/an-introduction-to-form-validation-methods/
19	http://www.webreference.com/programming/javascript/jkm4/index.html
20	http://www.it-c.dk/~lthorup/kuslich_jserror.html
21	OCR GCSE Computing for GCSE textbook
22	http://webapplog.com/php-vs-node-js/