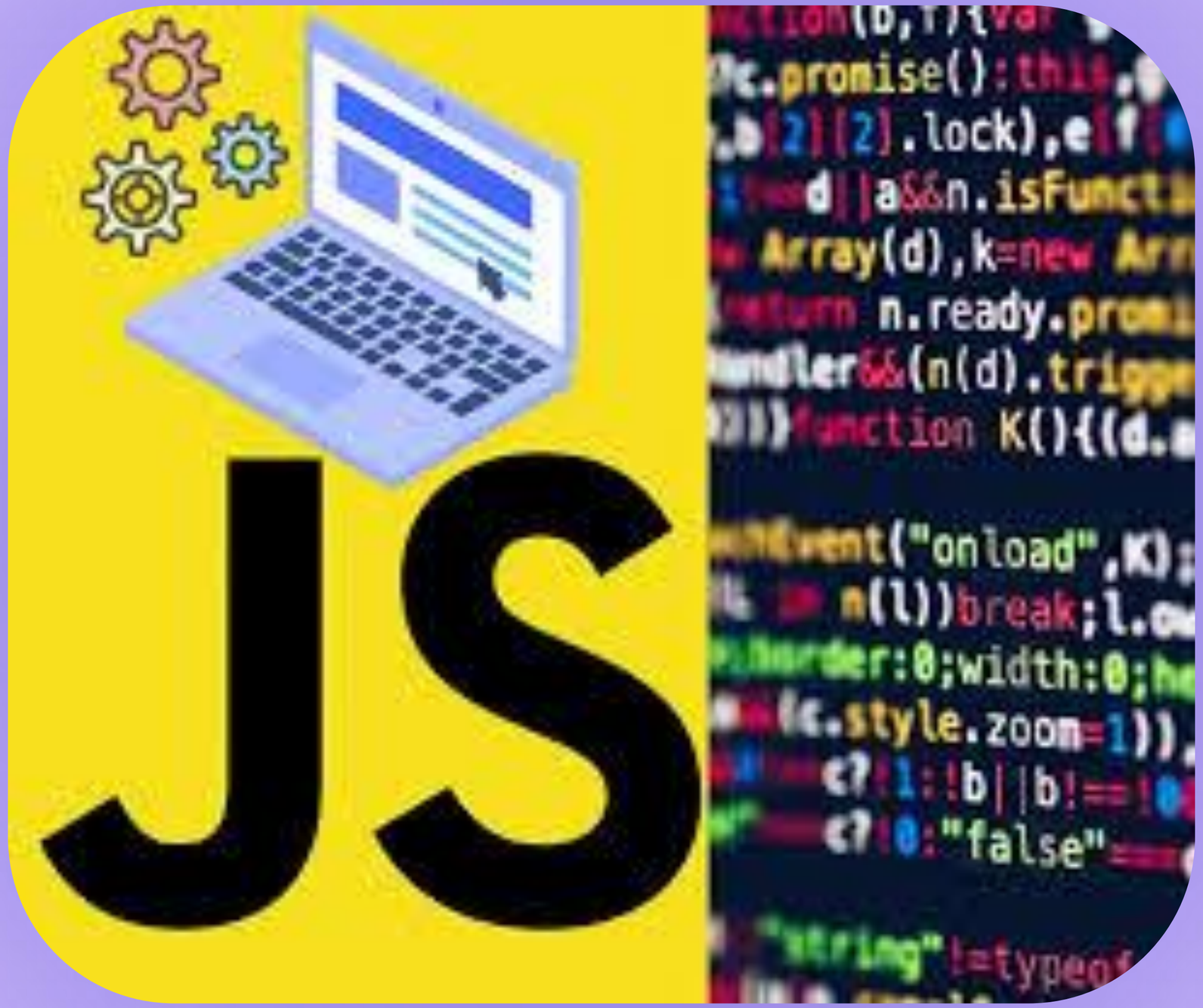
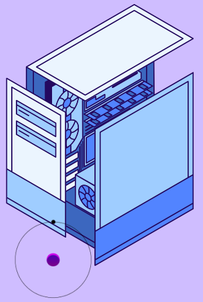
TLE/ICT 9

Fourth Quarter

Lesson 2


JS

Conditions, Controls and Loops



Conditional Statements

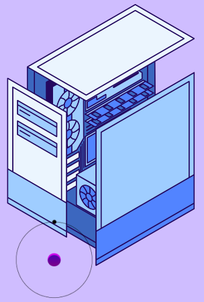
In JS, conditional statements perform different actions for different decisions.

If Statement

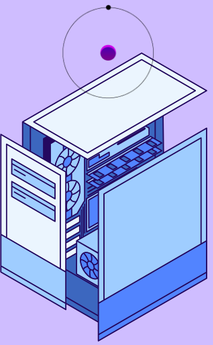
The “If Statement” is used to check or verify a condition and execute a set of statements only if the condition is true. This should be referred as a statement and not as a function.

The If statement is one of the most popular and most important conditional constructs in JS and in many other programming languages. This conditional statement construct evaluates a condition to True or False. It then runs a specific code depending on the result of this evaluation.

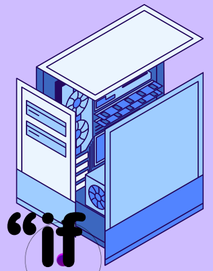
Try this Sample script



```
<!DOCTYPE html>
<head><title>IF</title>
<script type="text/javascript">
function check()
{if (document.example.answer.value=="Web Scripting")
{document.write("You answered correctly!");}}
</script></head>
<body>
<p>What is the title of this book?</p>
<form name="example">
<input type="text" name="answer" value="">
<input type="button" name="answerbutton" value="SUBMIT"
onClick=check()>
</form></body>
</html>
```



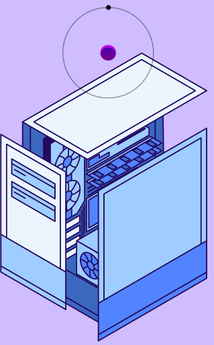
Nested If Statement

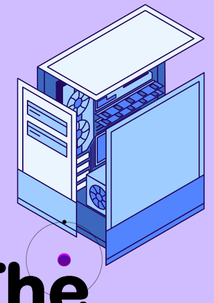


Nested If statement is like an “if statement” with another “if statement” just like a parent and a child. Nested if is used when “child” condition is only checked when the parent condition is true. For example, in purchasing a car, first the car color is verified if it looks good, only if it satisfies the parent condition which is the price and goes on to a more superior condition.

Syntax:

```
if (condition 1)
{
    if (condition 2)
    {
    //set of statements that will be executed
    }
}
```





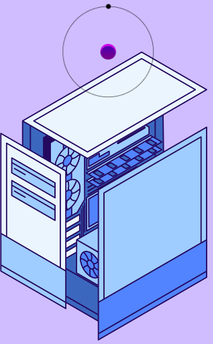
If-Else Statement

If-Else statement has also the same format as “if statement”. The only thing, is that an Else part is added to an If statement. So, if the condition satisfies the statement inside an If statement, part of the code will be executed. And Else statement inside Else part only will be executed.

The If-Else statement is similar to the If statement, except that we are giving an alternative instruction in case the arguments isn’t TRUE.

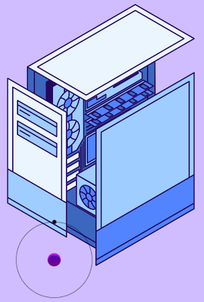
Syntax:

```
if(condition)
{//set of statements if condition satisfies}
else
{//set of statements if condition fails}
```



```
<!DOCTYPE html>
<head><title>If Else</title>
</head>
<body>
<script language="javascript">
var grade=60;
if(grade>=75)
{document.write("Passed");}
else
{document.write("Failed");}
</script>
</body>
</html>
```

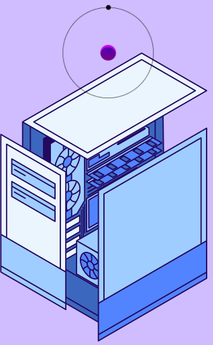
Try this Sample script

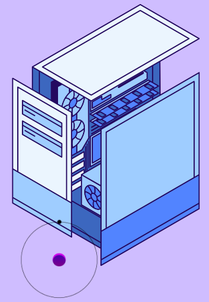


Note: In this example, the condition is to check if the variable grade equals(>=)75. If the condition is not satisfied as assigned in the initialization condition, then the Else part is executed. In this case, the variable grade was initialized at 60 which makes the condition false, hence the else portion is implemented giving the result "Failed".

Try this 2nd Sample script

```
<!DOCTYPE html>
<head><title>If Else</title>
<script type="text/javascript">
function check()
{
if (document.example.answer.value==”Web Scripting”)
{document.write(“You answered correctly!”);}
else
{document.write(“Sorry. Try again.”);}}
</script>
</head>
<body>
<p>What is the title of this book?</p>
<form name=“example”>
<input type=“text” name=“answer” value=“”>
<input type=“button” name=“answerbutton” value=“SUBMIT”
onClick=check()></form></body>
</html>
```



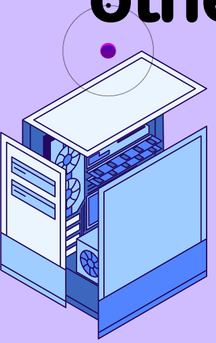


If Else If Statement

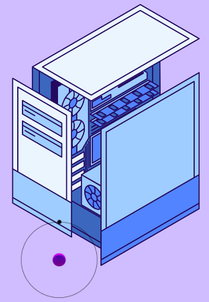
We don’t always evaluate just one condition. Sometimes more than one or multiple conditions must be evaluated.

This is possible with the If-Else-If statement. The name refers to an if statement that depends on another if statement.

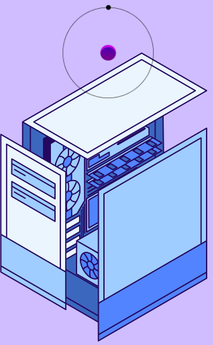
When you use the If Else If statement, you are simply telling the browser that is something is True, then do something else, if the other thing is True, then do something else, and so on.

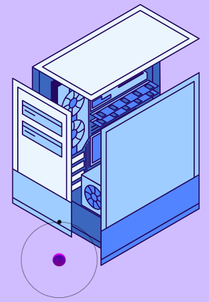


Try this Sample script



```
<!DOCTYPE html>
<head><title>If Else</title>
<script type="text/javascript">
function check()
{
if (document.example.answer.value==”Web Scripting”)
{document.write(“You answered correctly!”);}
else if (document.example.answer.value==“”)
{document.write(“You did not answer the question!”);}
else
{document.write(“Sorry. Try again!”);}}
</script>
</head>
<body>
<p>What is the title of this book?</p>
<form name=“example”>
<input type=“text” name=“answer” value=“”>
<input type=“button” name=“answerbutton” value=“SUBMIT” onClick=check()>
</form></body>
</html>
```

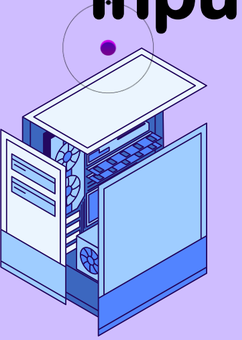


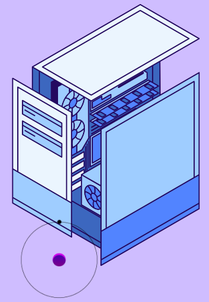


Switch Statement

In the previous example, we used the If Else If statement to test multiple conditions. What if there are a lot of options? The If Else If statement can be tedious. To address this issue, JS offers an easier way to implement multiple conditions, that is through the Switch statement.

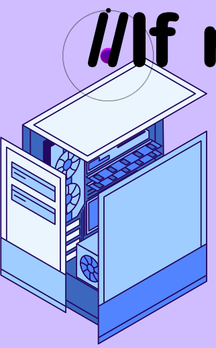
Switch statement is used when a condition may have multiple results and a different set of operation is done based on each result or input.



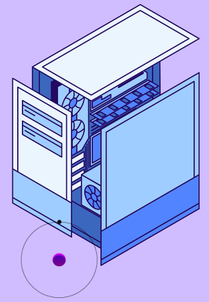


Syntax:

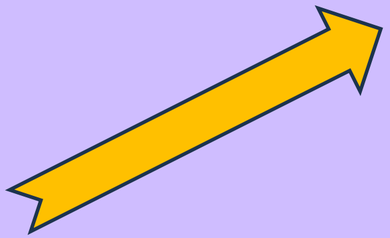
```
switch(condition)
{
case result1:
//operation for result 1

case result2:
//operation for result 2
.
.
default:
//If result belongs to none of the case specified. }
```



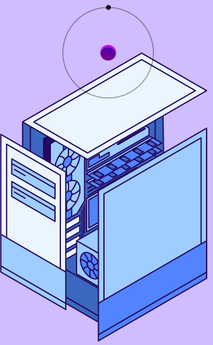
Try this Sample script

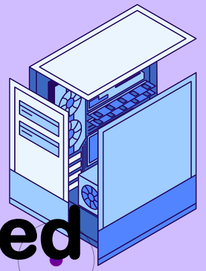


```
<!DOCTYPE html>
<head><title>Switch</title>
<script type="text/javascript">
function bookcheck()
{ var book=document.frmbook.cmbbook.value
switch (book)
{case "1": alert("J.R.R Tolkien")
break
case "2": alert("C.S. Lewis")
break
case "3": alert("J.K Rowling")
break
case "4": alert("Mark Twain")
break
default: alert("undetermined")}}
</script></head>
```

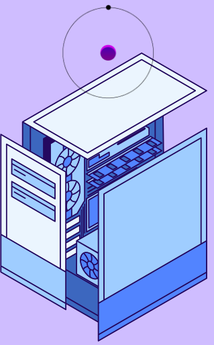


```
<body>
<p>Select a book below to find the name of the author</p>
<form name=formbook>
<select name=cmbbook>
<option value=1>Lord of the Rings</option>
<option value=2>Chronicles of Narnia
</option>
<option value=3>Harry Potter</option>
<option value=4>The Adventures of Huckleberry Finn</option></select>
<input type="button" name="combo"
value="SUBMIT" onClick=bookcheck()>
</form>
</body>
</html>
```





In the example, we started by creating a custom function named bookcheck(). Within that function we declared a variable declaring a variable called “book” and using that variable to get the value of the selected item in the combo box. We opened a Switch statement, passing in the variable we want to test which is “book”. This is is followed by a set of cases. The “break” construct after each case prevents the code from running into the next case. Prior to the closing bracket, we placed a default condition. This will be executed if none of the previous cases is true.



TLE-ICT
PERFORMANCE TASK

Submit your formal picture wearing your school uniform.

Requirements:

- Half Body Shot - until hips
- Background of the picture must be white or plain
- Facial details must be clear and recognizable
- Picture must be taken in eye-level position
- Ears must be visible (long hair)
- Shoulders must be visibly straight in the picture
- Cosmetics is allowed (light make up only)
- No wearing of eyeglasses
- Remove unecessary accessories (piercings,necklace etc)
- Long hair is not allowed (boys)
- Colored hair is not allowed (both)

SAVE IN .JPG FORMAT
FILENAME: CLASS NUMBER_SURNAME_GIVEN NAME
UPLOAD IN THE DRIVELINK PROVIDED

Photos will be used in Recognition Rites Slides Presentation.