Attendance

Please login to Canvas and “Check-In”

Attendance is not a part of your grade for this class. The university has mandated that we take attendance for all classes, face-to-face, online and hybrid, to assist in contact tracing should an outbreak of Covid-19 occur.

FOX
MIS

TEMPLE UNIVERSITY

Digital Systems

Logical Operators and Conditional Logic

FOX MIS

ROADMAP

START

Week 1:
Introduction & Systems Analysis
- Course Description
- Systems Thinking

Week 1:
Introduction to Process Mapping
- Systems & Processes
- Swim Lane Diagrams

- Max Labs 0- due
- Practice test - due

Week 2:
Digital Product Management & ERD

Week 2:
Introduction to Data Modeling

- Max Labs 1A/1B- due
- Max Labs 2A/2B due

Week 2:

Exam #1

10/30 – 11/1: Exam Availability

Week 3:
Information Systems
- ERP & CRM
- Data Analytics & SCM

- Lean IT #1 due

Week 4:
Platforms & Digital Business Models
- API's
- Cloud

Week 4:
Cybersecurity & AI
- Protection Protocols
- Artificial Intelligence

- Cybersecurity/AI assignment due
- Max Labs 3a/3b due

Week 4 :

Exam #2

11/13-11/5 Exam Availability

Week 5:
JavaScript Unit #1 & 2
- Hello World, Variables
- Input and Output
- Operator types
- Strings

Watch Lynda.com video – due
Code Academy due

Week 6:
JavaScript Unit #3&4
- Logical Operators
- Conditional Types
- Intro to Loops
- While and Do
- Writing the code
- Practice Coding Exam

Week 7:
HTML & CSS

- Coding Assignment -due
- Lean IT #2 due

Week 7:

Exam #3

12/8 – 12/ 9: Exam Availability

FINISH

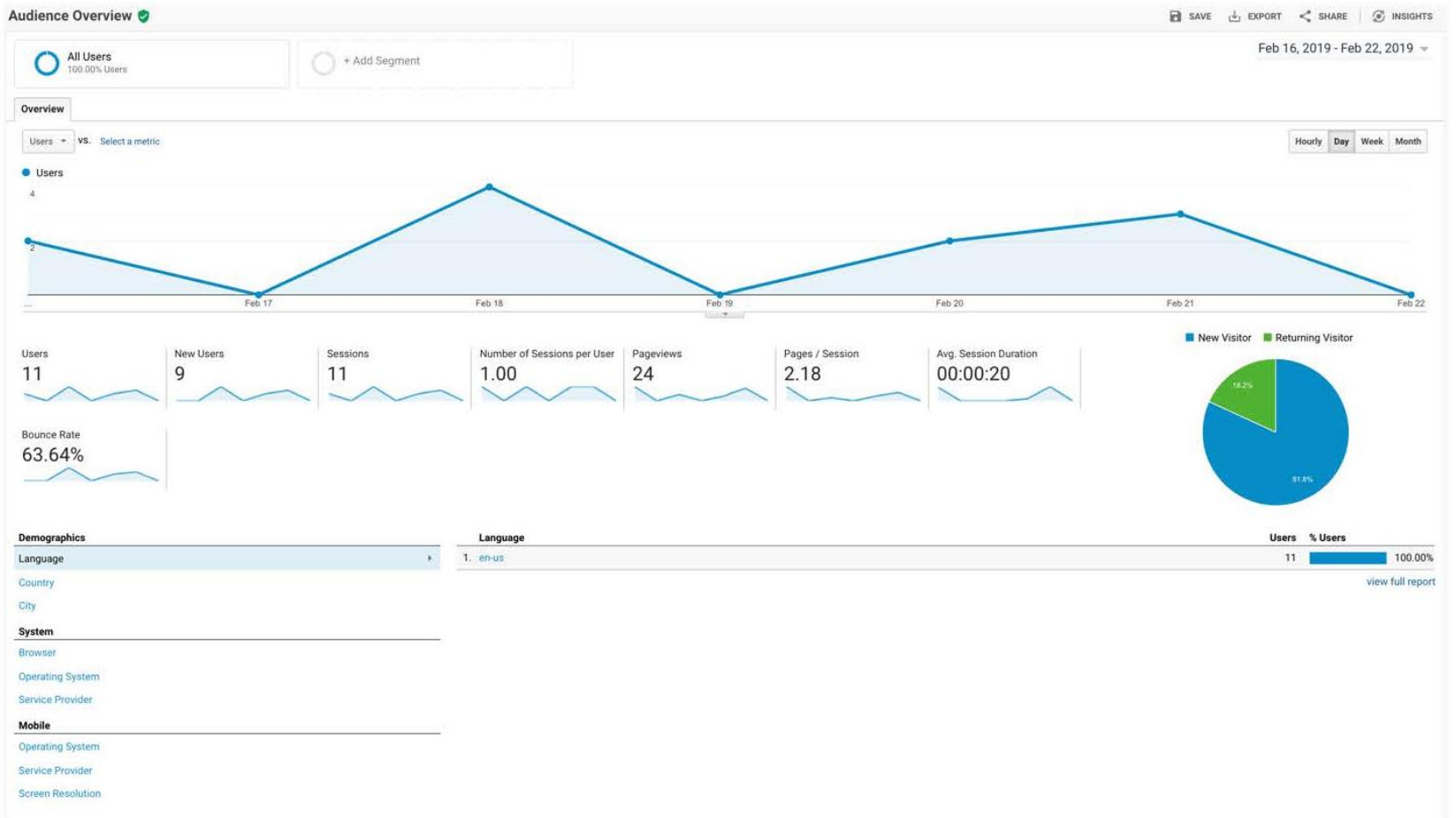
TIPS FROM MIS 2101 VIRTUAL HELPDESK

[The Only Way to Learn to Program is by Programming with Caroline Doyle]

TIP: SHARE YOUR SITE WITH FRIENDS AND FAMILY
https://community.mis.temple.edu/mcmartin/

Learn IT! #2

Part 3. – Google Analytics



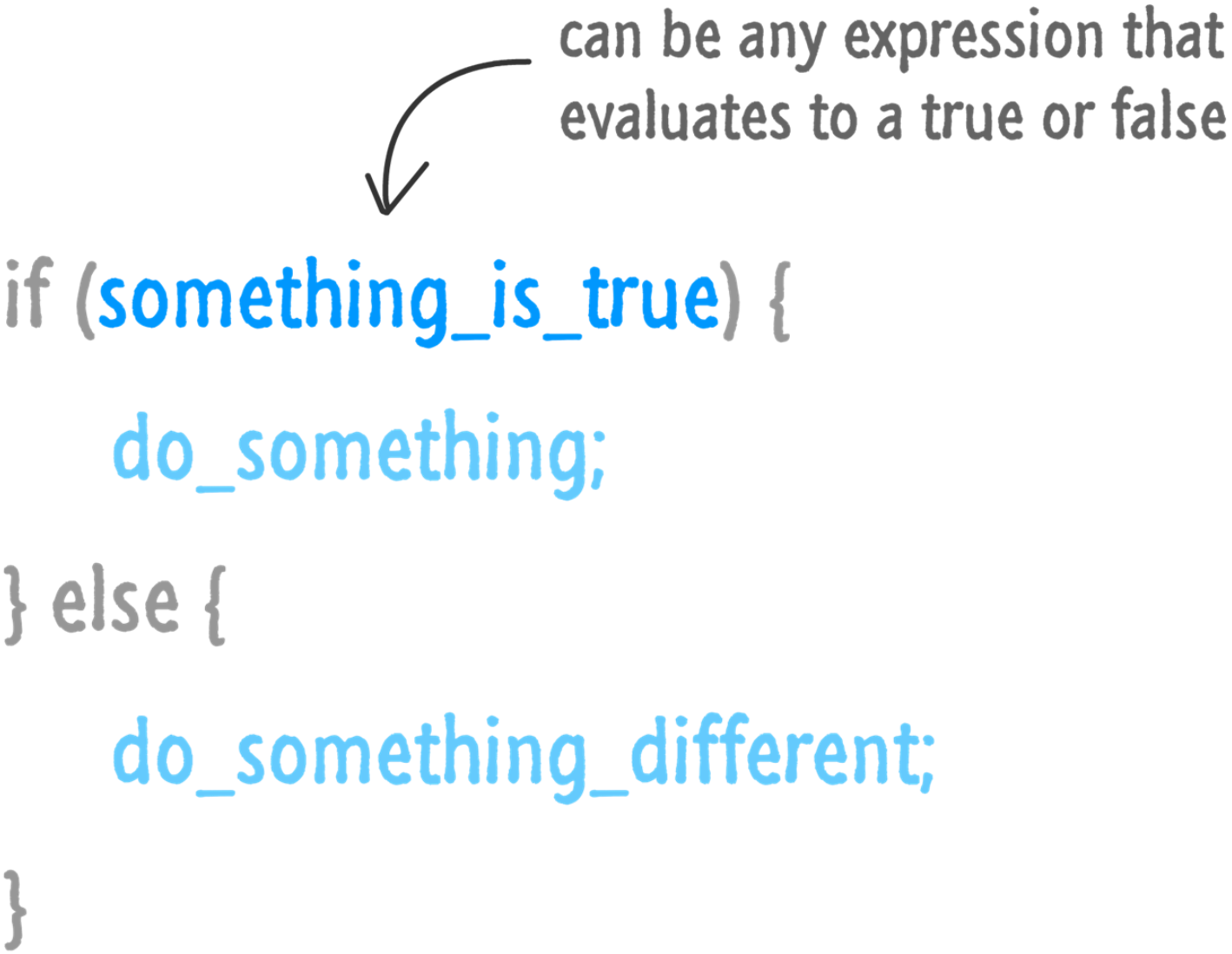
FOX MIS

If / Else Statements

Week 6

FOX
MIS

If and **Else** statements allow you to run some code based on whether a condition is **true** or **false**.



Source: JavaScript Absolute Beginner's Guide by Kirupa Chinnathambi

```
if (expression  operator  expression) {
    do_something;
} else {
    do_something_different;
}
```

Source: JavaScript Absolute Beginner's Guide by Kirupa Chinnathambi

Operator	When it is true...
==	If the first expression evaluates to something that is equal to the second expression.
>=	If the first expression evaluates to something that is greater or equal to the second expression
>	If the first expression evaluates to something that is greater than the second expression
<=	If the first expression evaluates to something that is lesser or equal to the second expression
<	If the first expression evaluates to something that is less than the second expression
!=	If the first expression evaluates to something that is not equal to the second expression
&&	If the first expression and the second expression both evaluate to true
||	If either the first expression or the second expression evaluate to true
!	Flips the value from false to true or true to false

Relational Operators

Logical Operators

Conditional Expressions

```
lastName == "Hopper"
testScore == 10

firstName != "Grace"
months != 0

testScore > 100
age < 18

distance >= limit
stock <= reorder_point

rate / 100 >= 0.1
```

Expressions evaluate to true or false.

The Order of Operations

- **Why is it important to remember PEMDAS while coding?**
  - **Because JavaScript follows these rules!**

Source: https://www.pinterest.com/pin/103442741288297 00/

Hello World!

(now with conditional logic)

Classroom Challenge

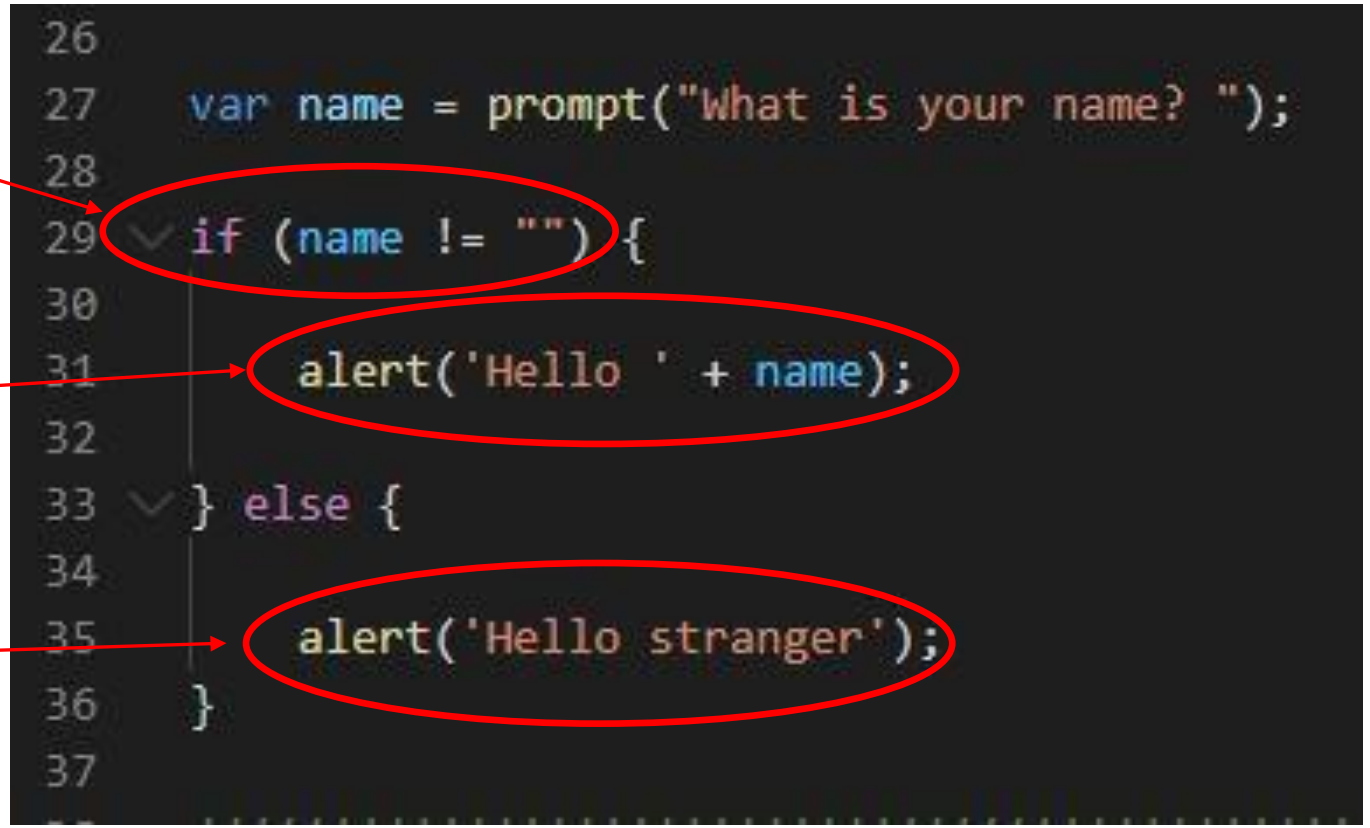
FOX
MIS

What gets displayed now?

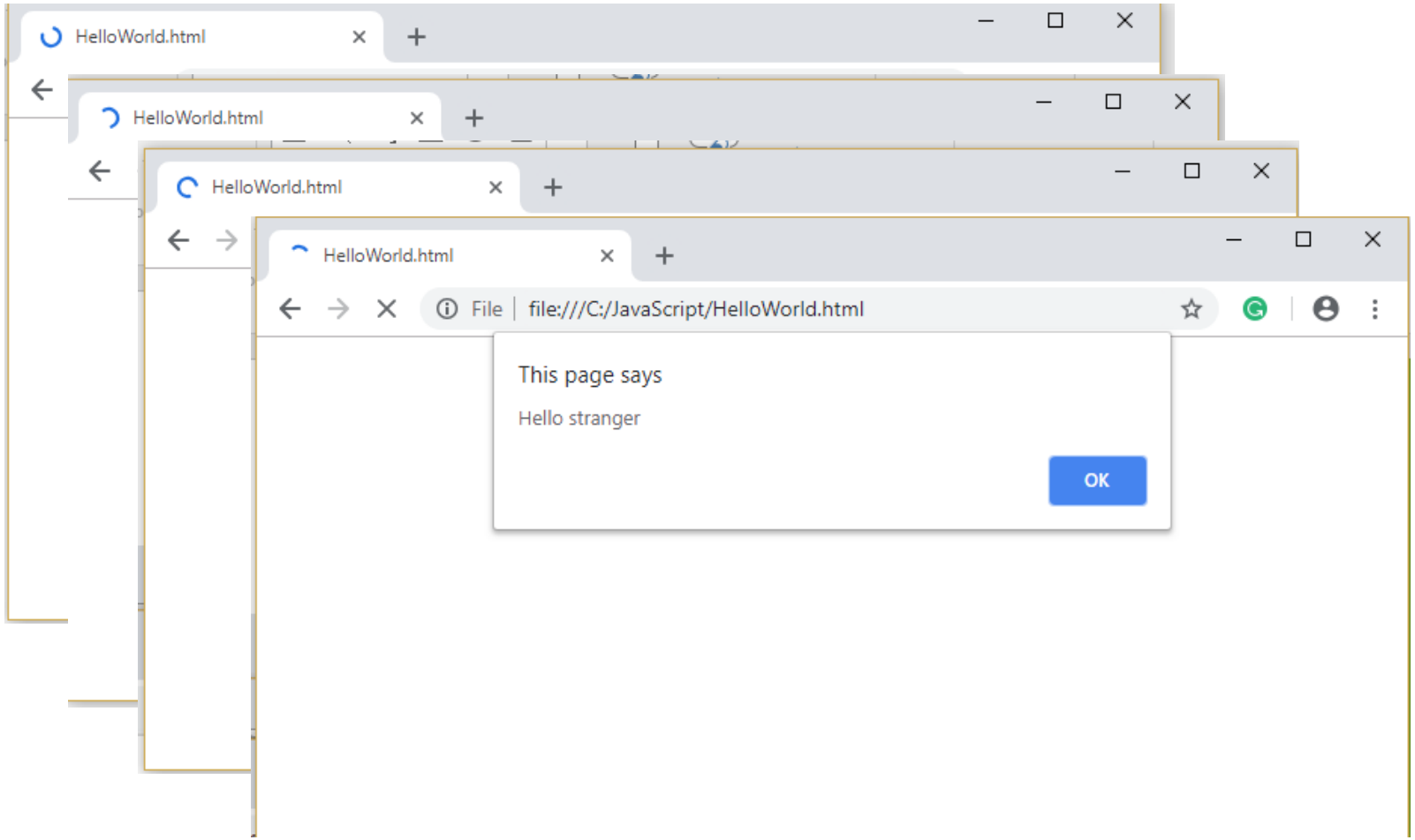
An “if” statement with a Boolean (true/false) expression

…what we do if the Boolean expression is true

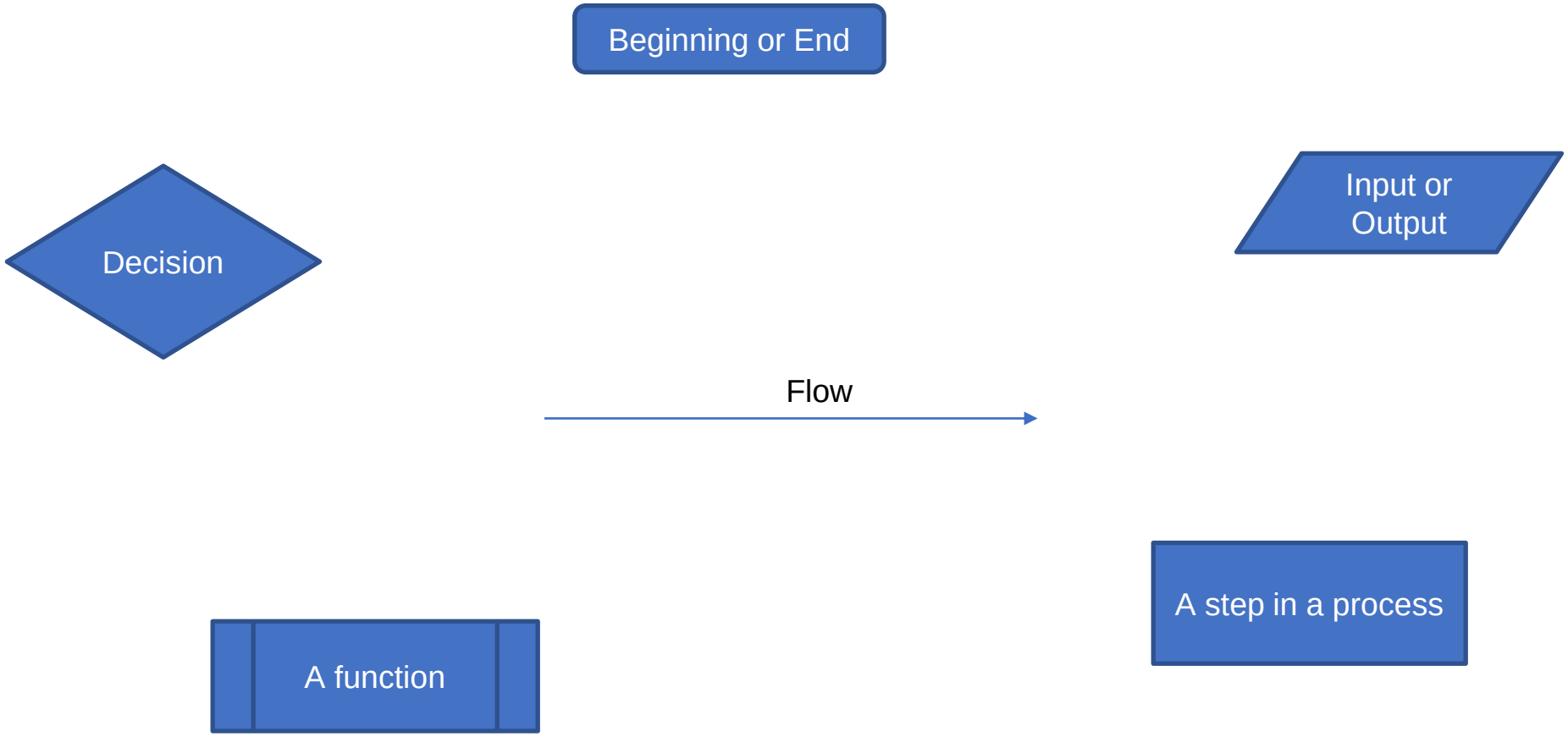
…what we do if the Boolean expression is false



Fancy Hello World! or Hello Stranger



Flowcharts

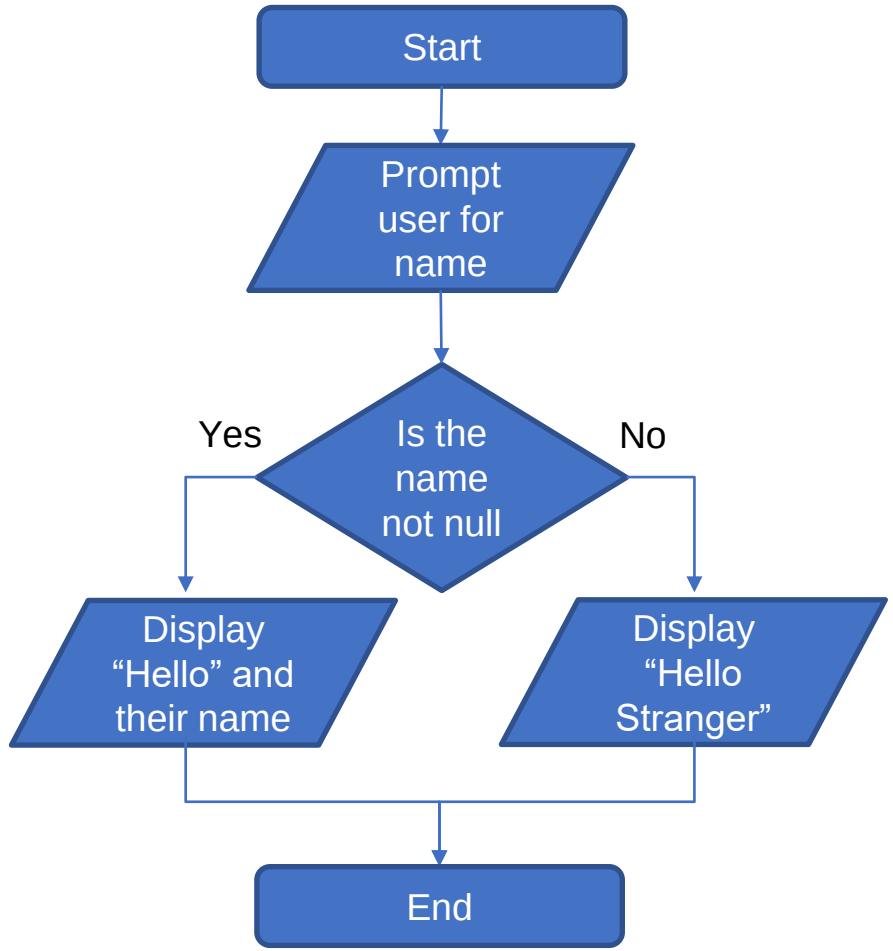

Beginning or End
Decision
Input or Output
Flow
A step in a process
A function

Hello World

Step #1 – Understand the Problem

Prompt the user for their name. If the user enters their name then display the message “Hello” and their name. If the user does not enter their name then display the message “Hello Stranger”

Step #2 – Develop the Algorithm



Practice, Practice, Practice

Open HelloWorld2.html and start coding!

```
26
27    var name = prompt("What is your name? ");
28
29    if (name != "") {
30
31        alert('Hello ' + name);
32
33    } else {
34
35        alert('Hello stranger');
36    }
37
```

Handy Boolean Expression

The syntax of the global isNaN method

```
isNaN(expression)
```

Examples of the isNaN() method

```
isNaN("Hopper")      // Returns true

isNaN("123.45")      // Returns false
```

isNaN() is a global method. The term “global” means it is available everywhere in your JavaScript code. Global methods are also sometimes called functions.

Conditional expressions with logical operators

Example 1: The AND operator

```
age > 17 && score < 70
```

Example 2: The OR operator

```
isNaN(rate) || rate < 0
```

Example 3: The NOT operator

```
!isNaN(age)
```

Expressions evaluate to true or false.

What do each of these expressions evaluate to?

Putting conditional expressions to work!

The syntax of the if statement

```
if ( condition-1 ) { statements }
[ else if ( condition-2 ) { statements }
  ...
  else if ( condition-n ) { statements } ]
[ else { statements } ]
```

An if statement

```
if ( age >= 18 ) {
    alert ("You may vote.");
}
```

Examples of using if/else clauses

An if statement with an else clause

```
if ( age >= 18 ) {
    alert ("You may vote.");
} else {
    alert ("You are not old enough to vote.");
}
```

An if statement with multiple else clauses

```
if ( isNaN(rate) ) {
    alert ("You did not provide a number for the rate.");
} else if ( rate < 0 ) {
    alert ("The rate may not be less than zero.");
} else if ( rate > 12 ) {
    alert ("The rate may not be greater than 12.");
} else {
    alert ("The rate is: " + rate + ".");
}
```

An Example

```
var speedLimit = 55;


function amISpeeding(speed) {
   if (speed >= speedLimit) {
       alert("Yes. You are speeding.");
   } else {
       alert("No. You are not speeding. What's wrong with you?");
   }
}
amISpeeding(53);
amISpeeding(72);
```

Your **if** and **else** statements can be nested to help you simulate more complex situations!

```
var xPos = 300;
var yPos = 150;

function sendWarning(x, y) {
    if ((x < xPos) && (y < yPos)) {
        alert("Adjust the position");
    } else {
        alert("Things are fine!");
    }
}

sendWarning(500, 160);
sendWarning(100, 100);
sendWarning(201, 149);
```

TIPS FROM MIS 2101 VIRTUAL HELPDESK

[Don’t Fall Behind with Jackson Randolph]

Time to code!

ICA 8: Logical Operators and Conditional Logic Part 1

GuessANumber Code

Diamond Peer Teacher
Sean Boyer

Guess A Number Coding Walkthrough

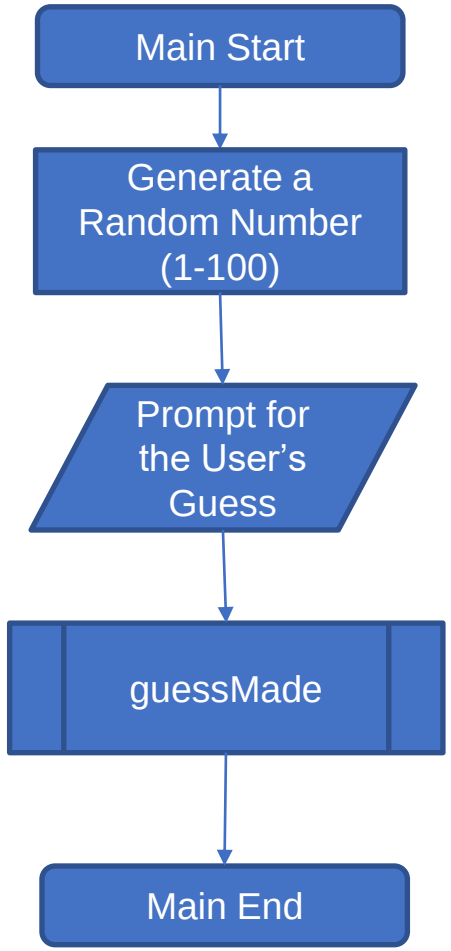
FOX
MIS

GuessANumber

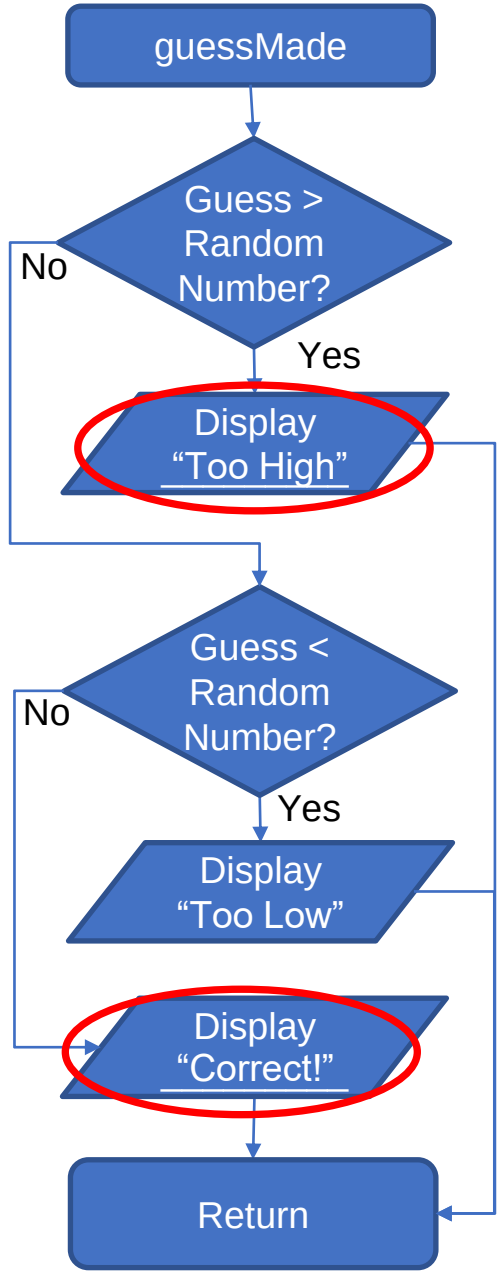
Step #1 – Understand the Problem

Update the GuessANumber program so that after the random number is generated and the user is prompted for a guess. If the guess is “Too high”, “Too low” or “Correct”, display the random number that was generated, the number that was guessed and an appropriate message that indicates if the guess was too high, too low or correct.

Step #2 – Develop the Algorithm



Step #3 – Write the Code



FOX MIS

the code: guess a number

```
<html>
<body>
<script>

function guessMade(numberGuessed){

        if (numberGuessed > randomNumber) {
                return 'Your guess is too high';
        } else if (numberGuessed < randomNumber) {
                return 'Your guess is too low';
        } else {
                return 'Your guess is correct!';
        }

}

var randomNumber = Math.floor(Math.random() * 100) + 1;
var numberGuessed = prompt('Guess a number from 1 to 100: ');

guessMade(numberGuessed);

</script>
</body>
</html>
```

1. The tags are already there: <html> for the html page, <body> for the content in the page, <script> for the code in the page.

2. Here we make the function called guessMade which takes in numberGuessed variable as a parameter.

3. This function does not return a value, instead based on the conditions, it will display text

4. Here we already set a random number using the random equation and get the user to take a guess.

5. The function is called with the guess as parameter and based on the conditions, the user will see the alert on the browser

6. We closed the tags

FOX MIS

“Challenges”!

- **GuessANumber – done!**
- **DayOfTheWeek**
- **AreasOfRectangles**
- **AgeClassifier**
- **RomanNumerals**
- **MassAndWeight**

Diamond Peer Teacher
Jack Granieri

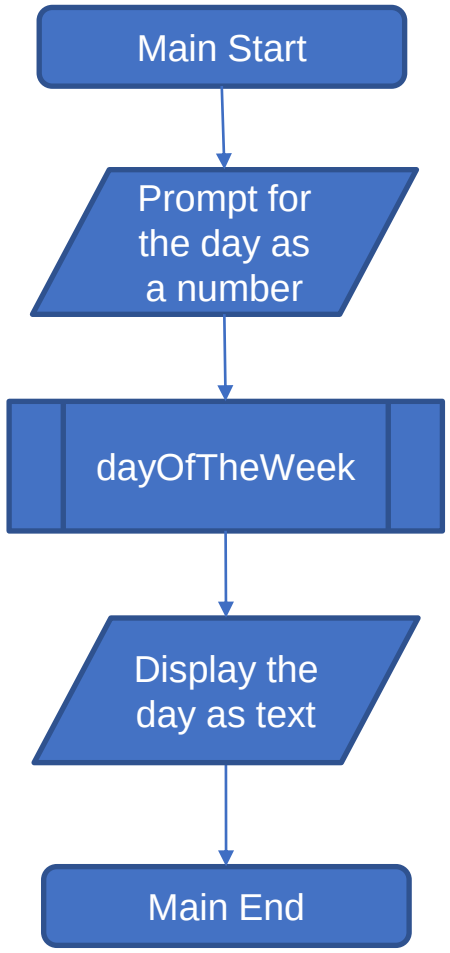
Day of the Week Coding Walkthrough

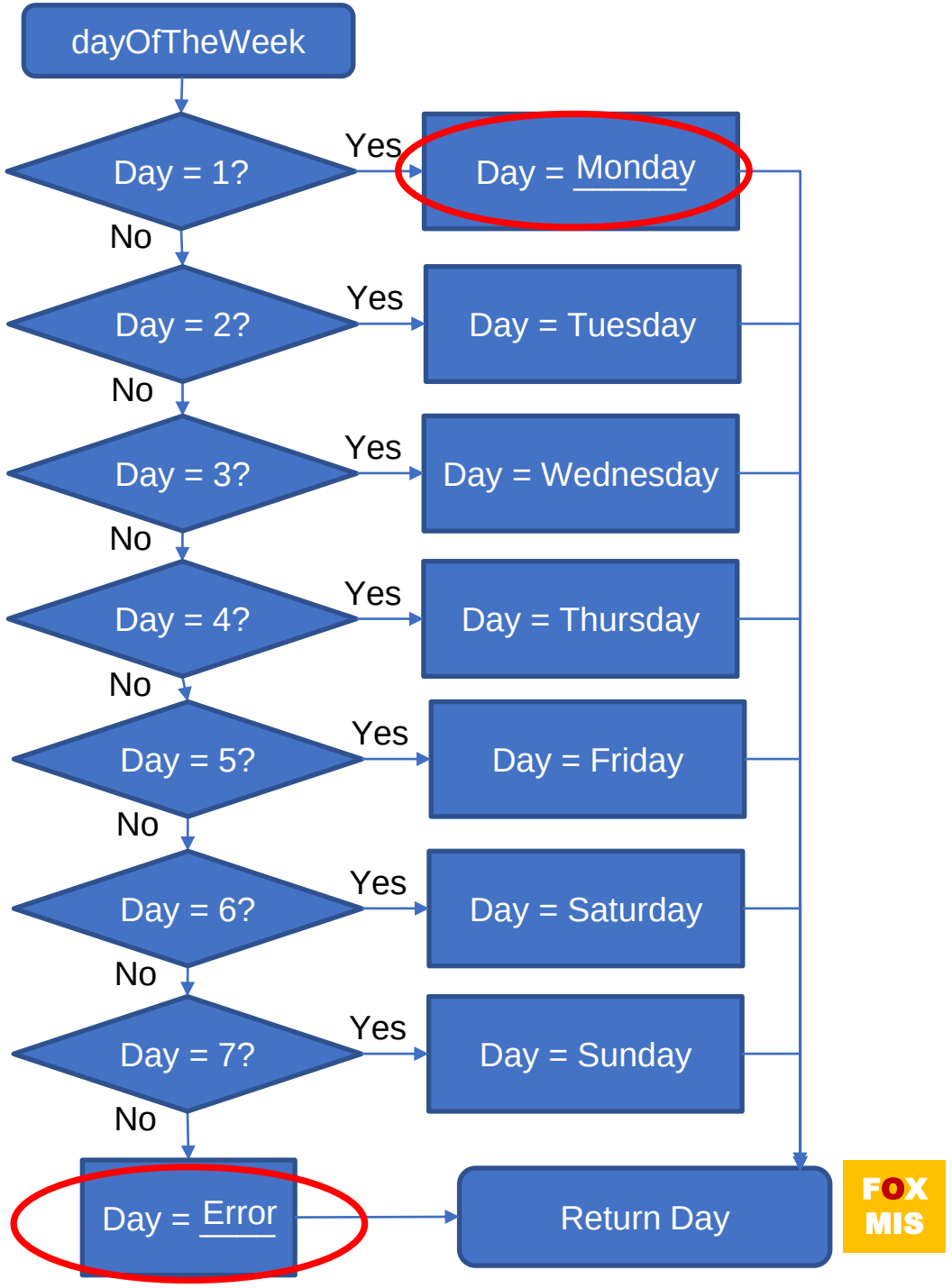
FOX
MIS

DayOfTheWeek

Step #1 – Understand the Problem

Write a program that asks the user for a number in the range of 1 through 7. The program should call a function which returns the corresponding day of the week, where 1 = Monday, 2 = Tuesday, 3 = Wednesday, 4 = Thursday, 5 = Friday, 6 = Saturday, and 7 = Sunday. The program should display an error message if the user enters a number that is outside the range of 1 through 7.





FOX MIS

the code: Day of the week

```
<html>
<body>
<script>
function dayOfTheWeek(day) {
        if (day == 1) {
                valueToReturn = 'Monday';
        } else if (day == 2) {
                valueToReturn = 'Tuesday';
        } else if (day == 3) {
                valueToReturn = 'Wednesday';
        } else if (day == 4) {
                valueToReturn = 'Thursday';
        } else if (day == 5) {
                valueToReturn = 'Friday';
        } else if (day == 6) {
                valueToReturn = 'Saturday';
        } else if (day == 7) {
                valueToReturn = 'Sunday';
        } else {
                valueToReturn = 'is not a valid day of the week';
        }
        return valueToReturn;
}
var day = parseInt(prompt('What day of the week (number)? '));
alert('Day ' + day + ' of the week is ' + dayOfTheWeek(day));
</script>
</body>
</html>
```

1. Make the tags: <html> for the html page, <body> for the content in the page, <script> for the code in the page.

2. Here we make the function called dayOfTheWeek which takes a number variable called day as the parameter.

3. This function does the conditional checking based on day and then returns a string, the name of the day

4. The value is returned from the function

5. The function is called with the guess as parameter and based on the conditions, the user will see the alert on the browser

6. Close the tags

FOX MIS

Homework

- **Review Riley's Ranking Calculator:**
  - **Let's look at the 3rd function together**
  - function calculateInvestorRanking(investmentAmount, annualIncome, assets, debts)

Loops

Week 6

FOX
MIS

TIPS FROM MIS 2101 VIRTUAL HELPDESK

[Programming is Easy but Requires a Different Way of Thinking with Andrew Smuszkiewicz]

- **So, what if you wanted to say Hello World ten times?**

Source: https://lh3.googleusercontent.com/ALTNv-dmwvGK0MTCa4XTQ9mCD1PMZZNfsaWSCt7PF9gbhpzol1hsHN5x-C6PXvjVldlkbek=s151

Source: https://lh3.googleusercontent.com/b7Cy0wGI1lYwR4mUZ7jpBLGxd0h0K_qBsK4zh61CXoEd2d1E5d4SV4KVh8a02KIH-tSy=s85

You could do this…

- **So, what happens if we run this?**

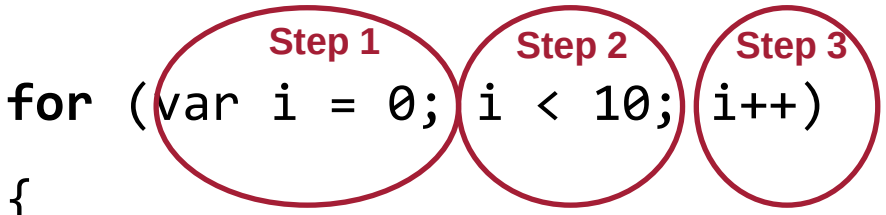
```
<!DOCTYPE html>
<html>
<body>

    <title> Steve Sclarow </title>
<script>

alert ("Hello World");
alert ("Hello World");
alert ("Hello World");
alert ("Hello World");
alert ("Hello World");
alert ("Hello World");
alert ("Hello World");
alert ("Hello World");
alert ("Hello World");
alert ("Hello World");

</script>
</body>
</html>
```

But wouldn't this be easier?




```
for (var i = 0; i < 10; i++)
{
    alert("Hello World!");
}
```

Enter the loop…

- A sequence of instructions that is repeated until a certain condition is reached.
- An operation is done, such as getting an item of data and changing it, and then some condition is checked such as whether a counter has reached a prescribed number.

Source: https://lh3.googleusercontent.com/ID3-M72vTnUEBpUxd_l835K2WC_ZUVjSkp7shlUbyX8jRDwPb2i7G-e7e9axmD19FbUEwg=s85

You will often want to repeat some code many **MANY** times.

Don’t do this!

```
saySomething();
saySomething();
saySomething();
saySomething();
saySomething();
saySomething();
saySomething();
saySomething();
```

You will often want to repeat some code many **MANY** times.

A **loop** will help you out.

Meet the loops!

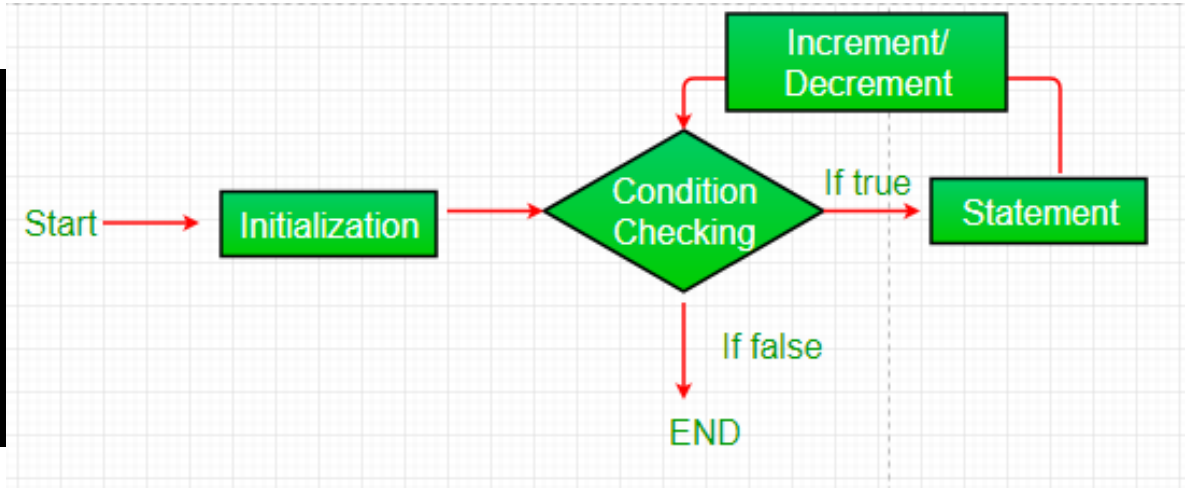
- There are three types of loops you can use to repeat some code:
  - **for** loop
  - **while** loop
  - **do...while** loop

These loops will be our focus

Source: https://lh3.googleusercontent.com/JSm2SSzaaB-sCwjg17mUWXPnca7FtOFItKZPsRby1DolGhyb_Kq_Nx7XB_AV4mMLluY8=s111

The `for` loop

```
for (var i = 0; i < 10; i++) {
 saySomething();
}
```



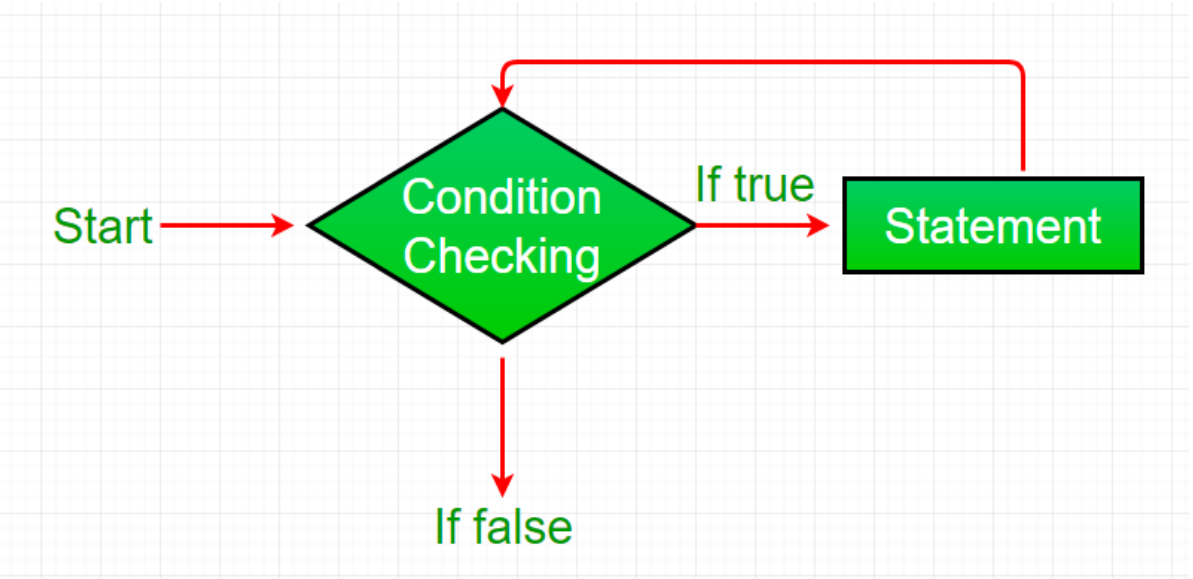
The **`for`** loop is all business. It requires you to define the looping conditions up front.

Source: https://lh3.googleusercontent.com/CAHXxrd2kB6nLkX4_NkIm4bqcvgyRmSxlps8LiKm8Ij8XuFwyOKTv2fDjHnkk1AwumdpJQ=s170

The `while` loop

```
var count = 0;


while (count < 10) {
    saySomething();
    count++;
}
```



The **`while`** loop will run until its looping condition evaluates to being **false**.

Source: https://lh3.googleusercontent.com/QYgoZkxEXZXRGVwkPq-q_D0YhQZ9DiDQnXty1LYS6nVAegwJPcxwzvVnJdDFtRgZAmkSKqo=s170

The while loop

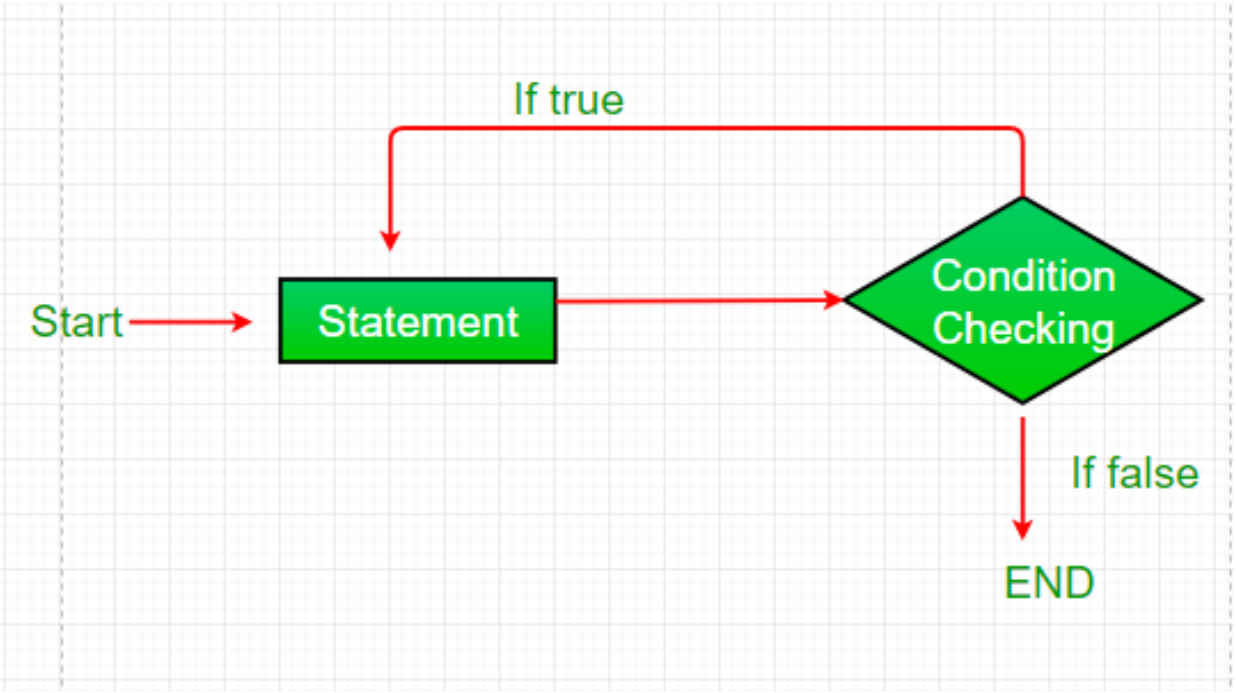
The **while** loop will run until its looping condition evaluates to being **false**.

The `do...while` loop

```
var count = 0;

do {
    count++;

    saySomething();
} while (count < 10);
```



The **`do...while`** loop is similar to the **`while`** loop we looked at earlier, but its looping condition is specified at the end.

Source: https://lh3.googleusercontent.com/OkdSdzKtxXuAXvIzm8DXa36ychqd1Ei3CgqDbrQKhhaSvaRow4DoUDSV6x-Vks5QGRk6=s152

Difference between while loop and do…while loop

Source: https://lh3.googleusercontent.com/79GT7HztldIm8wE-zJmIqNrvTH7NvOOq-u0vMjpZmdn3KKWKjF1KEwiKIBtpqb5ZyU7K4G4=s114

Challenges

- **BugCollector**
- **CaloriesBurned**
- **BudgetAnalysis**
- **DistanceTraveled**
- **C2FTable**
- **GuessANumber**

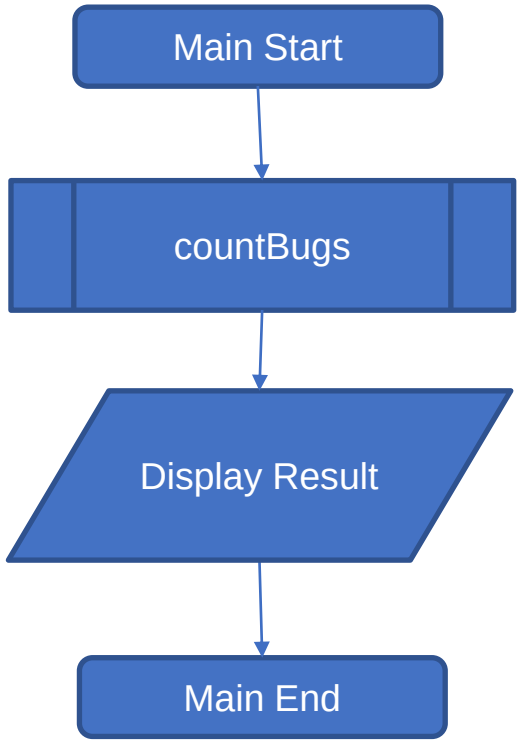
Diamond Peer Teacher
Patrick Jurgelewicz

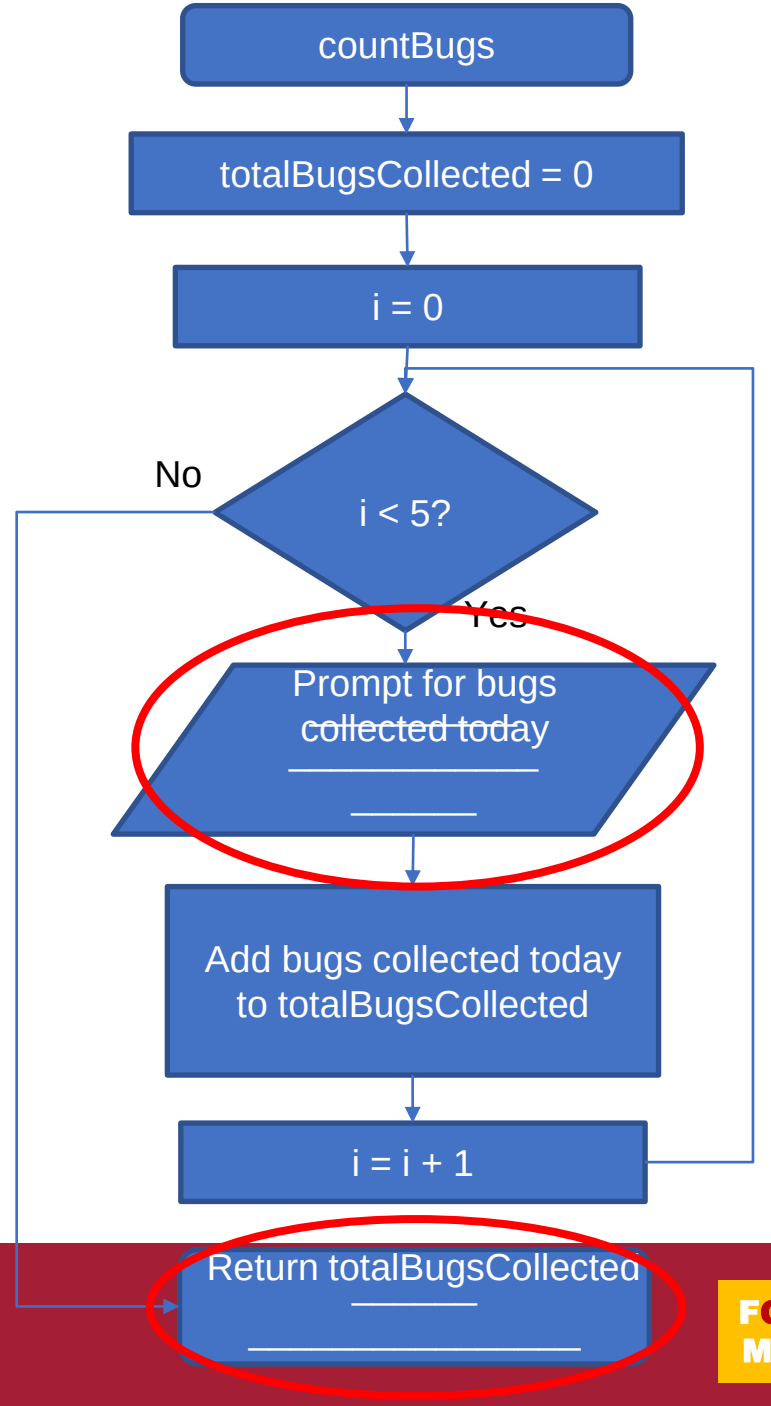
Bug Collector Coding Walkthrough

FOX
MIS

BugCollector

A bug collector collects bugs every day for five days. Write a function that keeps a running total of the number of bugs collected during the five days. The loop should ask for the number of bugs collected for each day, and when the loop is finished, the program should display the total number of bugs collected.





the code: bug collector

```
<html>
<body>
<script>

function countBugs() {

        var totalBugsCollected = 0;

        for (var i = 0; i < 5; i++ ) {
var bugsCollectedThisDay = parseInt(prompt('How many bugs were
collected on day ' + parseInt(i+1) + '? '));

totalBugsCollecetd = totalBugsCollecetd + bugsCollectedThisDay;
        }

        return totalBugsCollecetd;

}

alert('Over the past five days, ' + countBugs() + ' bugs were collected');
</script>
</body>
</html>
```

1. Make the tags: <html> for the html page, <body> for the content in the page, <script> for the code in the page.

2. Here we make the function called countBugs which has no parameter since we are taking input from the user inside the function.

3. This function does returns a total. So we make a variable that will increase the total every time the loop runs. The loop will ask user five time to enter the bug count and then add it to the total bug collected variable.

4. Here we return the total

5. The function is called with no parameter and the user will see the alert on the browser based on total bugs

6. Close the tags

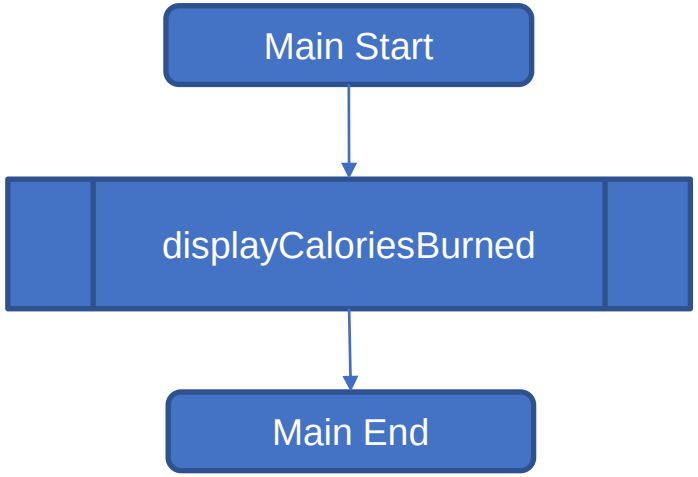
Diamond Peer Teacher
Sean Boyer

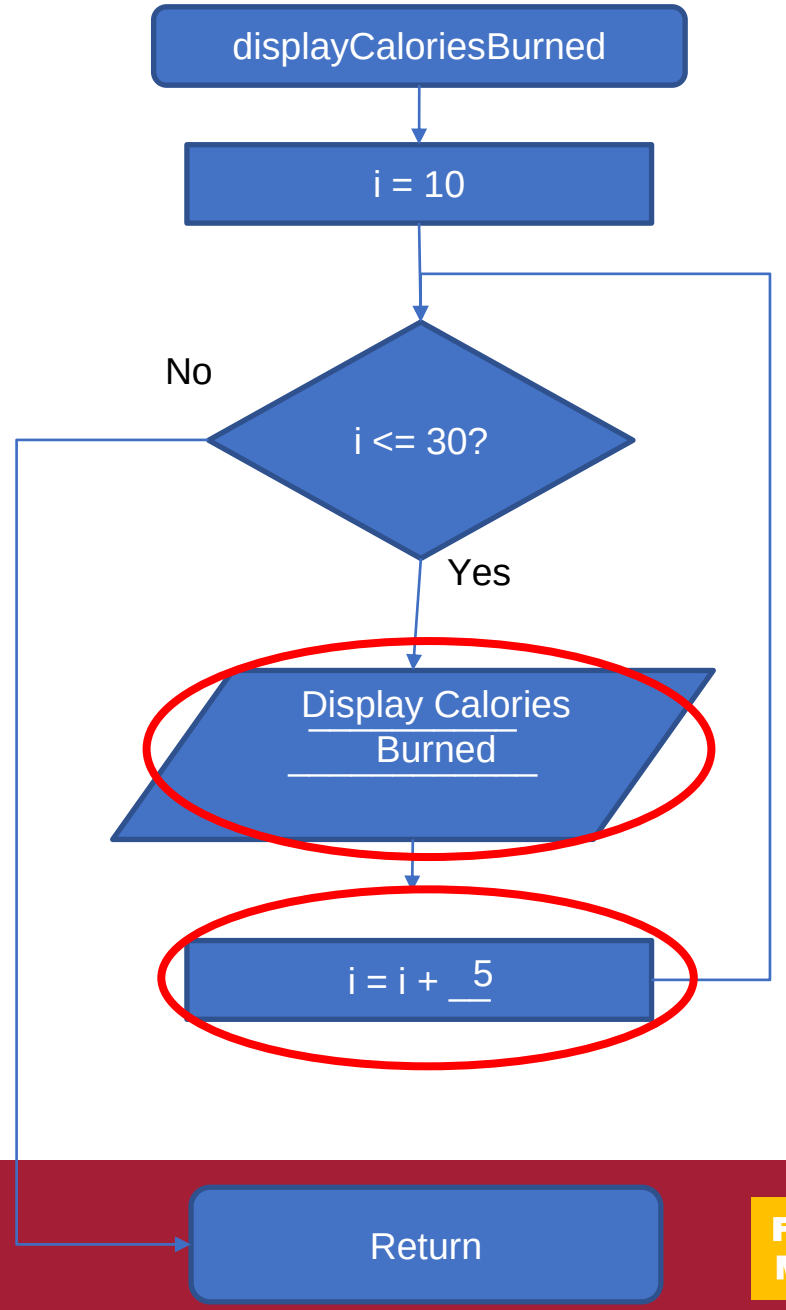
Calories Burned Coding Walkthrough

FOX
MIS

CaloriesBurned

Running on a particular treadmill you burn 4.2 calories per minute. Write a program that calls a function that uses a loop to display the number of calories burned after 10, 15, 20, 25, and 30 minutes.





the code: calories burned

```
<html>
<body>
<script>
function displayCaloriesBurned() {

for (var i = 10; i <= 30; i=i+5) {
alert('In ' + i + ' minutes, you will have burned ' +
i*4.5 + ' calories');
        }

}

displayCaloriesBurned();

</script>
</body>
</html>
```

1. Make the tags: <html> for the html page, <body> for the content in the page, <script> for the code in the page.

2. Here we make the function called displayCaloriesBurned with no parameters.

3. This function which is a for loop displaying the calories burned after 10, 15, 20, 25, and 30 minutes.

4. The function is called without parameters and the information is displayed in the browser

5. Close the tags

FOX MIS

Homework

- **Write the 1st function for Riley's Ranking Calculator:**
  - function totalAssets()
- **Write the 2nd function for Riley's Ranking Calculator:**
  - function totalDebt()

More to Come

Prepare with Readings & Videos before our next class!!!