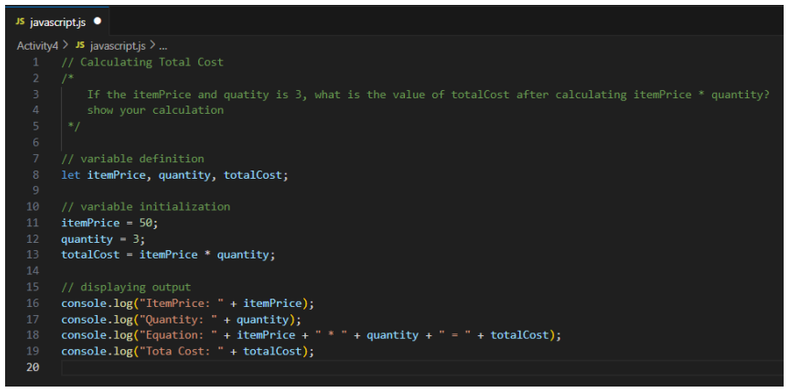
Activity 4: Combined Operators Questions:

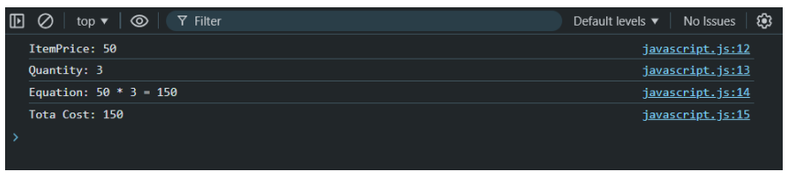
Instruction: Kindly read each number and show your code and output per question.

1. **Calculating Total Cost:**
   - If the itemPrice is 50 and quantity is 3, what is the value of totalCost after calculating itemPrice * quantity? Show your calculation.
     - **Source Code:**



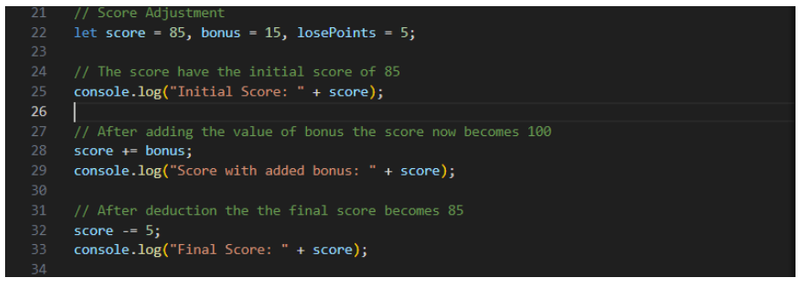
```
// Calculating Total Cost
/*
    If the itemPrice and quatity is 3, what is the value of totalCost after calculating itemPrice * quantity?
    show your calculation
*/

// variable definition
let itemPrice, quantity, totalCost;

// variable initialization
itemPrice = 50;
quantity = 3;
totalCost = itemPrice * quantity;

// displaying output
console.log("ItemPrice: " + itemPrice);
console.log("Quantity: " + quantity);
console.log("Equation: " + itemPrice + " * " + quantity + " = " + totalCost);
console.log("Tota Cost: " + totalCost);
```

     - **Output:**



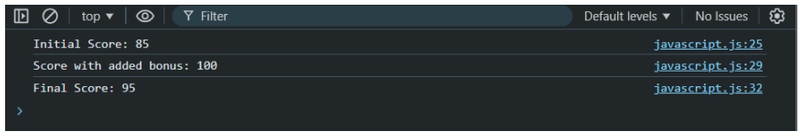
```
ItemPrice: 50                          javascript.js:12
Quantity: 3                            javascript.js:13
Equation: 50 * 3 = 150                 javascript.js:14
Tota Cost: 150                         javascript.js:15
```

2. **Score Adjustment:**
   - Starting with a score of 85, if you receive a bonus of 15 points and then lose 5 points, what is the final value of finalScore? How did you arrive at this number?
     - **Source Code:**



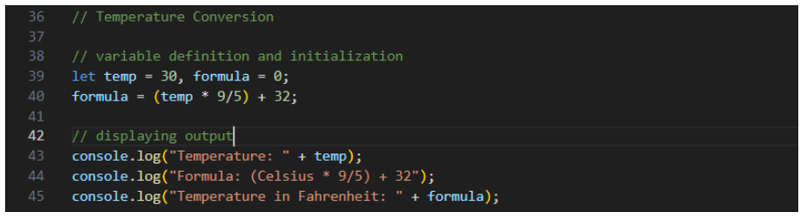
```
// Score Adjustment
let score = 85, bonus = 15, losePoints = 5;

// The score have the initial score of 85
console.log("Initial Score: " + score);

// After adding the value of bonus the score now becomes 100
score += bonus;
console.log("Score with added bonus: " + score);

// After deduction the the final score becomes 85
score -= 5;
console.log("Final Score: " + score);
```

     - **Output:**



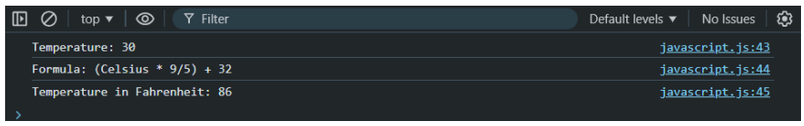
```
Initial Score: 85                      javascript.js:25
Score with added bonus: 100            javascript.js:29
Final Score: 95                        javascript.js:32
```

3. **Temperature Conversion:**
   - Given that the temperature is 30 degrees Celsius, what is the equivalent temperature in Fahrenheit using the formula (Celsius * 9/5) + 32? Calculate and provide the result.
     - **Source Code:**



```
// Temperature Conversion

// variable definition and initialization
let temp = 30, formula = 0;
formula = (temp * 9/5) + 32;

// displaying output
console.log("Temperature: " + temp);
console.log("Formula: (Celsius * 9/5) + 32");
console.log("Temperature in Fahrenheit: " + formula);
```

     - **Output:**

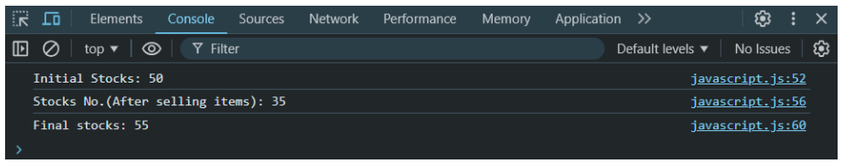


```
Temperature: 30                        javascript.js:43
Formula: (Celsius * 9/5) + 32          javascript.js:44
Temperature in Fahrenheit: 86          javascript.js:45
```

4. **Inventory Management:**
    - **If you start with itemsInStock = 50, sell 15 items, and then restock with 20 items, what will your final itemsInStock be? Show your calculations step-by-step.**
        - **Source Code:**

```
// Initial item stock
console.log("Initial Stocks: " + itemInStock);

// No. of items after selling 15 items
itemInStock -= sellItem;
console.log("Stocks No.(After selling items): " + itemInStock);

// Final no. of stocks
itemInStock += restockItem;
console.log("Final stocks: " + itemInStock);
```

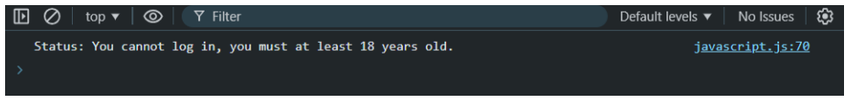
        - **Output:**



5. **Age Comparison:**
    - **If your age is 17, what message will be logged when checking if you are at least 18 years old? Explain why that message is logged.**
        - **Source Code:**

```
// variable definition and initialization
let age = 17;

// if else condition
if(age >= 18){
    console.log("Status: Logged")
} else {
    console.log("Status: You cannot log in, you must at least 18 years old.")
}

/*
    Explanation: If the age is greater than or equal to 18, the logged status will
    evaluate to true and display: "Status: Logged". Since the given age is 17, which
    is less than 18, the condition evaluates to false and displays the result:
    "Status: You cannot log in, you must be at least 18 years old."
*/
```

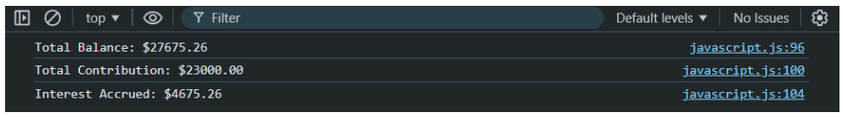
        - **Output:**



6. **Investment Growth with Monthly Contributions:**
    - **You start with an investment of $5000. Each month, you contribute an additional $300. If your investment grows at an annual interest rate of 6%, compounded monthly, what will your total balance be after 5 years? Provide the calculations for the interest accrued and total contributions**
        - **Source Code:**

```
let investment = 5000,
monthlyContribution = 300,
rate = 0.06,
years = 5,
months = years * 12,
monthlyRate = rate / 12;

// Calculate total balance with compounded interest
let totalBalance = investment * Math.pow((1 + monthlyRate), months);

// Add monthly contributions with compounding
for (let i = 1; i <= months; i++) {
    totalBalance += monthlyContribution * Math.pow((1 + monthlyRate), (months - i));
}
console.log("Total Balance: $" + totalBalance.toFixed(2));

// Calculate total contributions
let totalContribution = investment + (monthlyContribution * months);
console.log("Total Contribution: $" + totalContribution.toFixed(2));

// Calculate interest accrued
let interestAccrued = totalBalance - totalContribution;
console.log("Interest Accrued: $" + interestAccrued.toFixed(2));
```

- **Output:**



7. **Distance Travelled with Varying Speeds:**
   - **A cyclist travels at a speed of 20 km/h for the first 2 hours, then increases their speed to 25 km/h for the next 1.5 hours. After that, they take a 30-minute break. Finally, they ride at a speed of 15 km/h for the remaining distance of 10 km. Calculate the total distance traveled and total time spent on the journey**
     - **Source Code:**

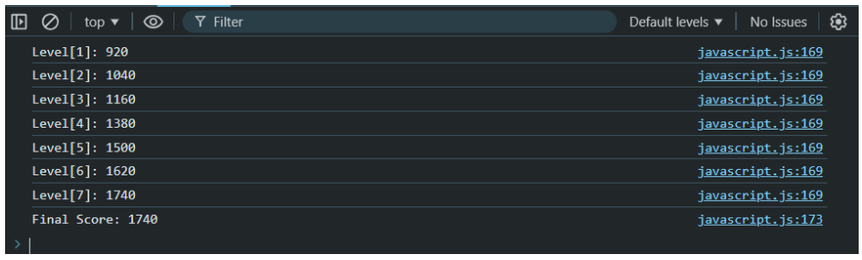
```
let speed = 0, time = 0, distance = 0, totalDistance = 0, totalTime = 0;

// First segment
speed = 20;
time = 2;
distance = speed * time;
totalDistance += distance;
totalTime += time;

console.log("First Segment: ")
console.log("Distance Travelled: " + totalDistance + " km");
console.log("Total Time Spent: " + totalTime + " km");
console.log("");

// Second Segment
speed = 25;
time = 1.5;
distance = speed * time;
totalDistance += distance;
totalTime += time;

console.log("Second Segment: ")
console.log("Distance Travelled: " + totalDistance + " km");
console.log("Total Time Spent: " + totalTime + " km");
console.log("");

//Break Segment
let breakTime = 0.5;
totalTime += breakTime;

console.log("Break Segment: ")
console.log("Distance Travelled: " + totalDistance + " km");
console.log("Total Time Spent: " + totalTime + " km");
console.log("");

// Third Segment
speed = 15;
distance = 10;
time = distance/speed;
totalTime += time;
totalDistance += distance;

console.log("Third Segment: ")
console.log("Final Distance Travelled: " + totalDistance + " km");
console.log("Final Total Time Spent: " + totalTime.toFixed(2) + " km");
```

     - **Output:**

8. **Enhanced Game Scoring System:**
    - **You begin with a score of 800. For every level completed (7 levels total), you gain 150 points and lose 30 points for penalties. Additionally, if you reach a score of 1200, you receive a bonus of 100 points. What will you final score be after all levels are completed?**
        - **Source Code:**

```
// initial score
let score = 800;
let bonusAward = false;
let level = 1;

for(level; level <= 7; level++){
    // for every level completed gain 150 and lose 30 points
    score += 150 - 30;

    // if you reach 1200 you will recieve a bonus of 100
    if(score >= 1200 && !bonusAward){
        score += 100;
        bonusAward = true;
    }


    // score every level
    console.log("Level[" + level + "]: " + score);
}

// final score
console.log("Final Score: " + score);
```

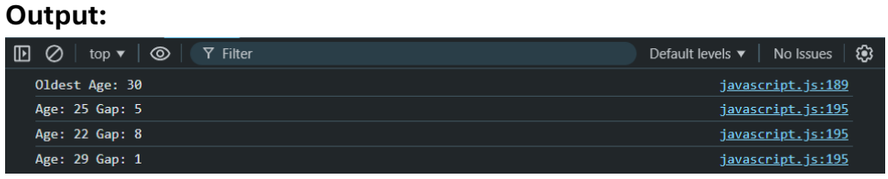
        - **Output:**



9. **Comparative Age Analysis:**
    - **Given the ages: age1 = 25, age2 = 30, age3 = 22, and age4 = 29, determine which person is the oldest and how much older they are than the others. Use comparison operators to assess the differences and log appropriate messages for each comparison.**
        - **Source Code:**

```
// define and intialize the data into array
let ages = [25, 30, 22, 29];
// get array length
let len = ages.length;
// assign the first array data
let oldest = ages[0];

// find the oldest age by looping the data
for(let i = 1; i < len; i++){
    if(ages[i] > oldest){
        oldest = ages[i]
    }
}

console.log("Oldest Age: " + oldest);

// compare each age to oldest age
for(let i = 0; i < len; i++){
    if(ages[i] < oldest){
        let gap = oldest - ages[i];
        console.log("Age: " + ages[i] + " Gap: " + gap);
    }
}
```

        - **Output:**



10. Dynamic Countdown Timer with Complex Conditions:

- **Starting with a count of 50, log the current count and decrement it. If the count is divisible by 5, you double the count before logging it. If the count is odd, subtract 1. How many times will you log a value before reaching 0, and what values will be logged during the countdown?**
    - **Source Code:**

```
// initial no. of count
let count = 50;
// initial no. of log count
let logCount = count;

// for loop the will iterate the count from 50 to 0
for (let i = count; i >= 0; i--) {
    // if i = 50 decrement the logCount by 1
    if(i == 50){
        --logCount;
    } else { // else if count is no longer 50 proceed to validate other condition
        // if count is divisible by 5 double the logged count
        if (i % 5 == 0 && !i == 0) {
            logCount *= 2;
        }
        // if count is an odd no. subtract 1
        if (i % 2 != 0) {
            --logCount;
        }
    }
    // no. of counts and no. of log counts
    console.log("Counts: " + i + " | " + "Log Counts: " + logCount);
}
```

Log No.	Repetition
49	1
48	2
47	2
93	2
92	2
91	1
182	1
181	2
180	2
359	2
358	2
357	1
714	1
713	2
712	2
1423	2
1422	2
1421	1
2842	1
2841	2
2840	2
5679	2
5678	2
5677	1
11354	1
11353	2
11352	2
22703	2
22702	2
22701	2

- **Output:**

```
Counts: 50 | Log Counts: 49                                    javascript.js:220
Counts: 49 | Log Counts: 48                                    javascript.js:220
Counts: 48 | Log Counts: 48                                    javascript.js:220
Counts: 47 | Log Counts: 47                                    javascript.js:220
Counts: 46 | Log Counts: 47                                    javascript.js:220
Counts: 45 | Log Counts: 93                                    javascript.js:220
Counts: 44 | Log Counts: 93                                    javascript.js:220
Counts: 43 | Log Counts: 92                                    javascript.js:220
Counts: 42 | Log Counts: 92                                    javascript.js:220
Counts: 41 | Log Counts: 91                                    javascript.js:220
Counts: 40 | Log Counts: 182                                   javascript.js:220
Counts: 39 | Log Counts: 181                                   javascript.js:220
Counts: 38 | Log Counts: 181                                   javascript.js:220
Counts: 37 | Log Counts: 180                                   javascript.js:220
Counts: 36 | Log Counts: 180                                   javascript.js:220
Counts: 35 | Log Counts: 359                                   javascript.js:220
Counts: 34 | Log Counts: 359                                   javascript.js:220
Counts: 33 | Log Counts: 358                                   javascript.js:220
Counts: 32 | Log Counts: 358                                   javascript.js:220
Counts: 31 | Log Counts: 357                                   javascript.js:220
Counts: 30 | Log Counts: 714                                   javascript.js:220
Counts: 29 | Log Counts: 713                                   javascript.js:220
Counts: 28 | Log Counts: 713                                   javascript.js:220
Counts: 27 | Log Counts: 712                                   javascript.js:220
Counts: 26 | Log Counts: 712                                   javascript.js:220
Counts: 25 | Log Counts: 1423                                  javascript.js:220
Counts: 24 | Log Counts: 1423                                  javascript.js:220
Counts: 23 | Log Counts: 1422                                  javascript.js:220
Counts: 22 | Log Counts: 1422                                  javascript.js:220
Counts: 21 | Log Counts: 1421                                  javascript.js:220
Counts: 20 | Log Counts: 2842                                  javascript.js:220
Counts: 19 | Log Counts: 2841                                  javascript.js:220
Counts: 18 | Log Counts: 2841                                  javascript.js:220
Counts: 17 | Log Counts: 2840                                  javascript.js:220
Counts: 16 | Log Counts: 2840                                  javascript.js:220
Counts: 15 | Log Counts: 5679                                  javascript.js:220
Counts: 14 | Log Counts: 5679                                  javascript.js:220
Counts: 13 | Log Counts: 5678                                  javascript.js:220
Counts: 12 | Log Counts: 5678                                  javascript.js:220
Counts: 11 | Log Counts: 5677                                  javascript.js:220
Counts: 10 | Log Counts: 11354                                 javascript.js:220
Counts: 9 | Log Counts: 11353                                  javascript.js:220
Counts: 8 | Log Counts: 11353                                  javascript.js:220
Counts: 7 | Log Counts: 11352                                  javascript.js:220
Counts: 6 | Log Counts: 11352                                  javascript.js:220
Counts: 5 | Log Counts: 22703                                  javascript.js:220
Counts: 4 | Log Counts: 22703                                  javascript.js:220
Counts: 3 | Log Counts: 22702                                  javascript.js:220
Counts: 2 | Log Counts: 22702                                  javascript.js:220
Counts: 1 | Log Counts: 22701                                  javascript.js:220
Counts: 0 | Log Counts: 22701                                  javascript.js:220
```