

## Conditional.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Conditional Statements</title>
  <link rel="stylesheet" href="style.css">
  <link
href="https://fonts.googleapis.com/css2?family=Bungee&family=Rubik:wght@400;500;7
00&display=swap" rel="stylesheet">
</head>
<body>
  <div class="grid-container">
    <header class="main-header">
      <h1>Conditional Statements</h1>
      <nav class="main-nav">
        <ul>
          <li class="links"><a href="conditional.html">Home</a></li>
          <li class="links"><a href="prime.html">Prime</a></li>
          <li class="links"><a href="lists.html">Lists</a></li>
        </ul>
      </nav>
    </header>

    <main class="main-content">
      <div class="split-container">
        <div class="left-pane">
          <h2>Age Checker</h2>
          <p>Check if a person is an adult or minor based on age.</p>

          <div class="age-checker">
            <input type="number" id="ageInput" placeholder="Enter
your age " required>
            <button id="checkAgeBtn" onclick="checkAge()">Check
Age</button>
          </div>
        </div>
        <div class="right-pane">
          <h2>Even or Odd?</h2>
          <p>Check if a number is even or odd.</p>

          <div class="number-checker">
```

```

        <input type="number" id="numberInput" placeholder="Enter
a number" required>
        <button id="checkNumberBtn"
onclick="checkEvenOdd()">Check Number</button>
    </div>
</div>
</div>
</main>

<footer class="main-footer">
    <p>&copy; 2025 John Rich Nicolas. All rights reserved.</p>
</footer>

</div>
<script src="script.js"></script>
</body>
</html>

```

Prime.html

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Prime Numbers</title>
    <link rel="stylesheet" href="style.css">
    <link
href="https://fonts.googleapis.com/css2?family=Bungee&family=Rubik:wght@400;500;7
00&display=swap" rel="stylesheet">
</head>
<body>
    <div class="grid-container">
        <header class="main-header">
            <h1>Prime Numbers</h1>
            <nav class="main-nav">
                <ul>
                    <li class="links"><a href="conditional.html">Home</a></li>
                    <li class="links"><a href="prime.html">Prime</a></li>
                    <li class="links"><a href="lists.html">Lists</a></li>
                </ul>
            </nav>
        </header>
    </div>

```

```

    <!-- BUTTON-->
    <div class="prime-button-wrapper">
        <button id="showGridBtn" onclick="generatePrimeGrid()">Show Prime
Grid</button>
        <footer class="main-footer">
            <p>&copy; 2025 John Rich Nicolas. All rights reserved.</p>
        </footer>
    </div>

    <!-- GRID goes here-->
    <div class="prime-grid" id="primeGrid"></div>

    <script src="script.js"></script>
</body>
</html>

```

Lists.html

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0"/>
    <title>Lists of Numbers</title>
    <link rel="stylesheet" href="style.css" />
    <link
href="https://fonts.googleapis.com/css2?family=Bungee&family=Rubik:wght@400;500;7
00&display=swap" rel="stylesheet" />
</head>
<body>
    <div class="grid-container">
        <header class="main-header">
            <h1>Lists of Numbers</h1>
            <nav class="main-nav">
                <ul>
                    <li class="links"><a href="conditional.html">Home</a></li>
                    <li class="links"><a href="prime.html">Prime</a></li>
                    <li class="links"><a href="lists.html">Lists</a></li>
                </ul>
            </nav>
        </header>

        <main class="main-content">
            <div class="split-container">

```

```

    <!-- Left Pane: Multiples of 3 -->
    <div class="left-pane">
        <h2>Multiples of 3</h2>
        <p>Generates a list of the first 10 multiples of 3.</p>
        <div class="multiples-generator">
            <button id="generateMultiplesBtn"
onclick="generateMultiples()">Generate Multiples</button>
        </div>
        <ul id="multiplesList"></ul>
    </div>

    <!-- Right Pane: Divisible by 5 -->
    <div class="right-pane">
        <h2>Divisible by 5</h2>
        <p>Shows the last 10 numbers divisible by 5 between 1-100.</p>
        <div class="divisible-generator">
            <button id="generateDivisiblesBtn"
onclick="generateDivisibleByFive()">Generate Divisibles</button>
        </div>
        <ul id="divisiblesList"></ul>
    </div>
</main>
</div>

<script src="script.js"></script>
</body>
</html>

```

Style.css

```

body {
    font-family: 'Rubik', sans-serif;
    background-color: #1e1e2f;
    color: #f1f1f1;
    padding: 20px;
}

.grid-container {
    display: grid;
    grid-template-areas:
        "header"
        "main"
        "footer";
}

```

```
    grid-template-columns: 1fr;
    grid-template-rows: auto 1fr auto;
    gap: 20px;
    max-width: 1200px;
    margin: auto;
    padding: 20px;
}

.main-footer{
    grid-area: footer;
    text-align: center;
}

/* Header Section */
.main-header {
    grid-area: header;
    display: flex;
    justify-content: space-between;
    align-items: center;
    flex-wrap: wrap;
    background-color: #4FC3F7;
    color: #1e1e2f;
    padding: 30px;
    border-radius: 1.5rem;
    border-bottom: 5px solid #FF6F61;
    box-shadow: 0 6px 16px rgba(0, 0, 0, 0.5);
    margin: 0;
}

.main-header h1 {
    font-family: 'Bungee', cursive;
    font-size: 2.5rem;
    letter-spacing: 1px;
    text-transform: uppercase;
}

.main-nav ul {
    list-style: none;
    display: flex;
    gap: 20px;
    padding: 20px 0;
    margin: 0;
}

.links a {
    text-decoration: none;
```

```
    color: #1e1e2f;
    font-weight: 600;
    transition: color 0.3s ease;
}

.links a:hover {
    color: #FF6F61;
}

/* Main Content Section */

.split-container {
    display: flex;
    gap: 20px;
}

.left-pane,
.right-pane {
    text-align: center;
    flex: 1;
}

.left-pane {
    background-color: #FF6F61;
    padding: 20px;
    border-radius: 1rem;
    color: #1e1e2f;
}

.right-pane {
    background-color: #4FC3F7;
    padding: 20px;
    border-radius: 1rem;
    color: #1e1e2f;
}

/* Age Section & number checker*/
.age-checker, .number-checker {
    display: flex;
    flex-direction: column;
    gap: 12px;
    align-items: center;
}
```

```
.age-checker input[type="number"], .number-checker input[type="number"] {
  padding: 12px 16px;
  border: none;
  border-radius: 8px;
  width: 200px;
  font-size: 1rem;
  box-shadow: 0 2px 6px rgba(0, 0, 0, 0.1);
}

/* Button Styles for conditional html*/
.age-checker button{
  padding: 12px 20px;
  background-color: #007BFF;
  color: white;
  border: none;
  border-radius: 8px;
  font-size: 1rem;
  cursor: pointer;
  transition: background-color 0.3s ease;
}

.age-checker button:hover {
  background-color: #0056b3;
}

.number-checker button {
  padding: 12px 20px;
  background-color: #FF6F61;
  color: white;
  border: none;
  border-radius: 8px;
  font-size: 1rem;
  cursor: pointer;
  transition: background-color 0.3s ease;
}

.number-checker button:hover {
  background-color: #e7210f;
}

/* Grid for prime numbers file */
#primeGrid {
  display: grid;
  grid-template-columns: repeat(4, 1fr);
  gap: 15px;
  margin-top: 20px;
}
```

```
padding: 10px;
}

.grid-item {
  background-color: #FF6F61;
  color: white;
  font-size: 1.5rem;
  display: flex;
  align-items: center;
  justify-content: center;
  border-radius: 10px;
  height: 100px;
  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.2);
}

.prime-button-wrapper {
  text-align: center;
  margin: 20px auto;
}

#showGridBtn {
  padding: 12px 24px;
  font-size: 1.1rem;
  border: none;
  border-radius: 8px;
  background-color: #007BFF;
  color: white;
  cursor: pointer;
  box-shadow: 0 3px 6px rgba(0, 0, 0, 0.2);
  transition: background-color 0.3s ease;
}

#showGridBtn:hover {
  background-color: #0056b3;
}

/* FOR LISTS */
.multiples-generator button, .divisible-generator button {
  padding: 12px 20px;
  background-color: #007BFF;
  color: white;
  border: none;
  border-radius: 8px;
  font-size: 1rem;
  cursor: pointer;
}
```



```

    transition: background-color 0.3s ease;
}

ul {
    list-style-type: none;
    padding: 0;
    margin: 20px auto;
    width: fit-content;
    text-align: center;
}

li {
    font-size: 1.5rem;
    margin: 4px 0;
}

```

Script.js

```

// Age Checker Script
function checkAge() {
    const input = document.getElementById('ageInput').value.trim();
    const age = parseFloat(input);

    if (input === "" || isNaN(age)) {
        alert("Enter valid number or input!");
        return;
    }

    if (age >= 18) {
        alert("You are legally an adult!");
    } else if (age >= 13 && age < 18) {
        alert("You are a teenager.");
    } else if (age >= 2 && age < 13) {
        alert("You are a child.");
    } else if (age <= 1) {
        alert("Invalid age input.");
    }
}

// Number Checker if odd or even
function checkEvenOdd() {
    const input = document.getElementById('numberInput').value.trim();
    const number = parseFloat(input);
    //checker if empty or di number
    if (input === "" || isNaN(number)) {
        alert("That is not a valid number!");
    }
}

```

```

        return;
    }
    if (number % 2 === 0) {
        alert("The number is even.");
    } else {
        alert("The number is odd.");
    }
}

//script for the prime 4x4 grid
//eto function pang check kung prime number
function isPrime(n) {
    if (n < 2) return false;
    for (let i = 2; i <= Math.sqrt(n); i++) {
        if (n % i === 0) return false;
    }
    return true;
}

//eto function pang generate ng 4x4 grid ng prime numbers
function generatePrimeGrid() {
    const container = document.getElementById('primeGrid');
    container.innerHTML = ""; // clear existing grid

    let count = 0;
    let num = 2;

    while (count < 16) {
        if (isPrime(num)) {
            const box = document.createElement('div');
            box.className = 'grid-item';
            box.textContent = num;
            container.appendChild(box);
            count++;
        }
        num++;
    }
}

// Function to handle the button click in lists
function generateMultiples() {
    const list = document.getElementById('multiplesList');
    list.innerHTML = "";

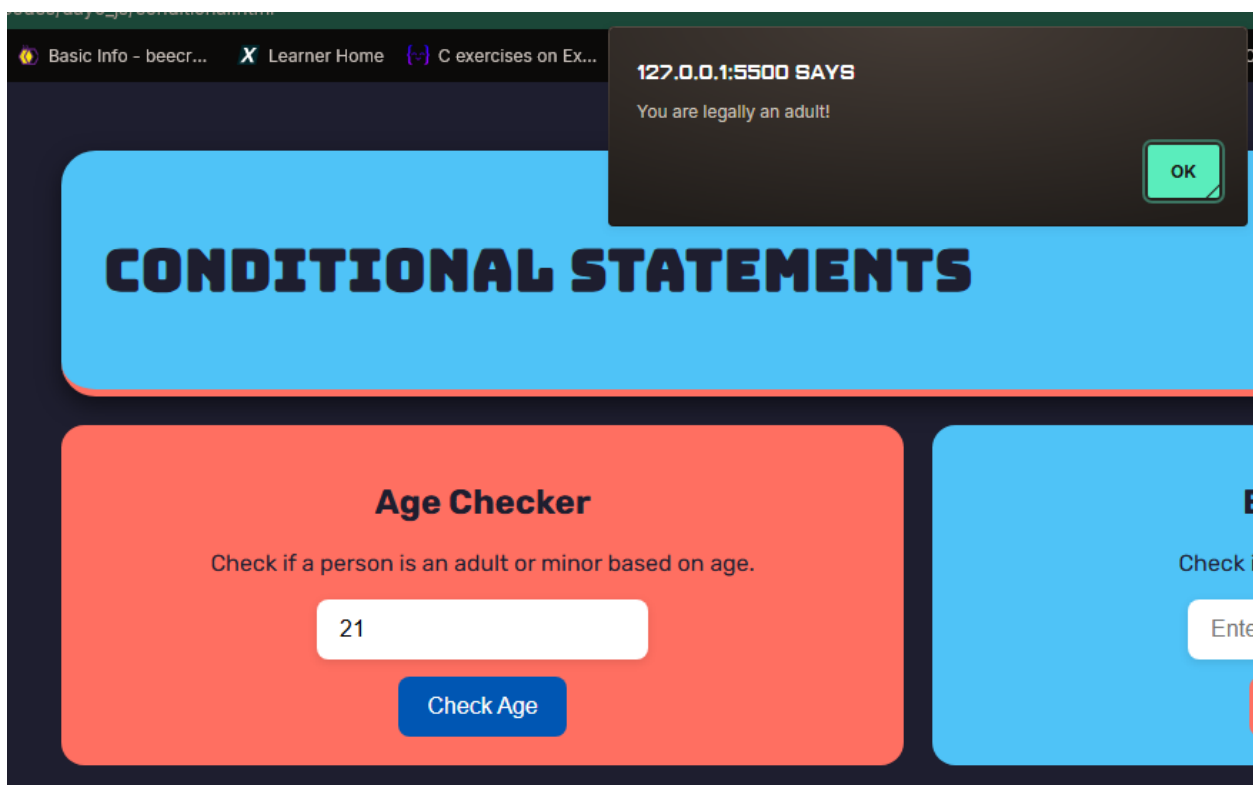
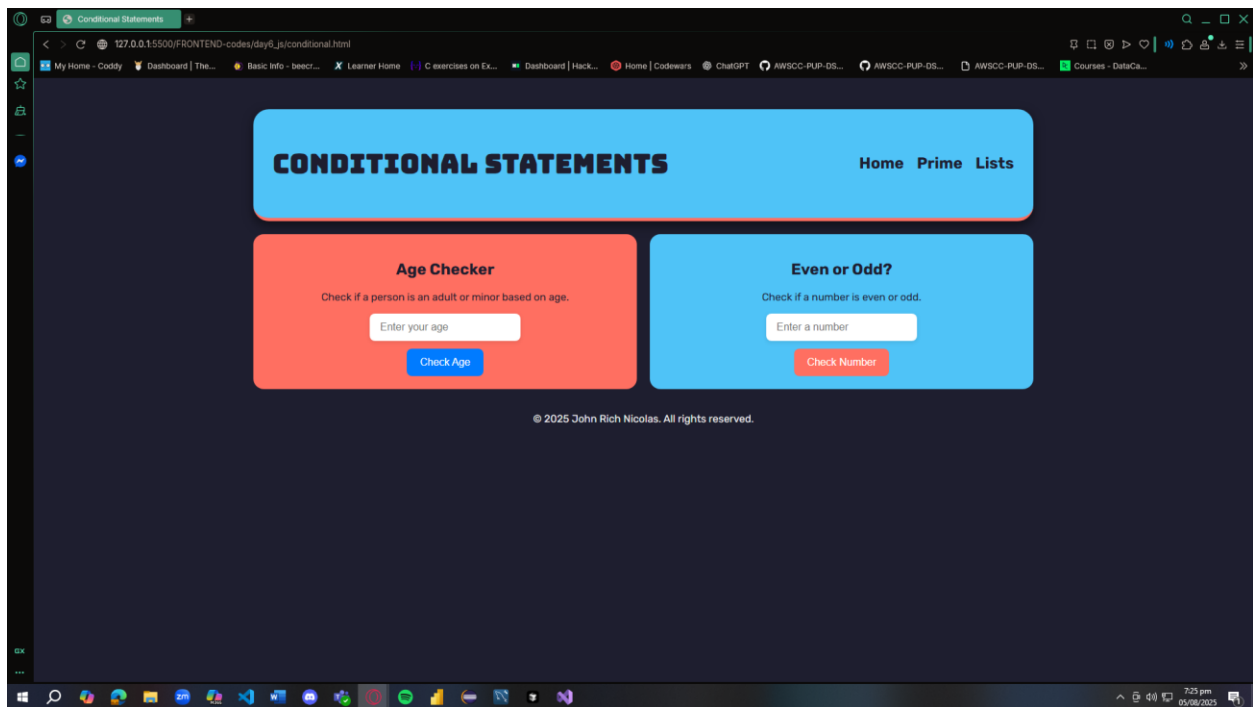
    //loloop lang 1 - 10 kase first 10 multiples lang naman of 3
    for (let i = 1; i <= 10; i++) {

```

```
    const li = document.createElement('li');
    li.textContent = `${i * 3}`;
    list.appendChild(li); //lalagay lang content
  }
}

function generateDivisibleByFive() {
  const list = document.getElementById('divisiblesList');
  list.innerHTML = "";
  let count = 0;
  for (let i = 100; i >= 1 && count < 10; i--) { //dahil last 10 daw so start sa
100
    if (i % 5 === 0) {
      const li = document.createElement('li');
      li.textContent = i;
      list.appendChild(li);
      count++;
    }
  }
}
```

## OUTPUT



127.0.0.1:5500 SAYS

You are a teenager.

OK

# CONDITIONAL STATEMENTS

## Age Checker

Check if a person is an adult or minor based on age.

Check Age

Check i

Ente

127.0.0.1:5500 SAYS

You are a child.

OK

# CONDITIONAL STATEMENTS

## Age Checker

Check if a person is an adult or minor based on age.

Check Age

Check i

Ente

Basic Info - beecr...X Learner HomeC exercises on Ex...

127.0.0.1:5500 SAYS  
Invalid age input.

OK

# CONDITIONAL STATEMENTS

## Age Checker

Check if a person is an adult or minor based on age.

1

Check Age

Check

E

Basic Info - beecr...X Learner HomeC exercises on Ex...

127.0.0.1:5500 SAYS  
Invalid age input.

OK

# CONDITIONAL STATEMENTS

## Age Checker

Check if a person is an adult or minor based on age.

-232

Check Age

Check

Ent

## ODD/PRIME NUMBERS

127.0.0.1:5500 SAYS  
The number is odd.  
OK

STATEMENTS Home Prime Lists

Even or Odd?  
Check if a number is even or odd.  
1  
Check Number

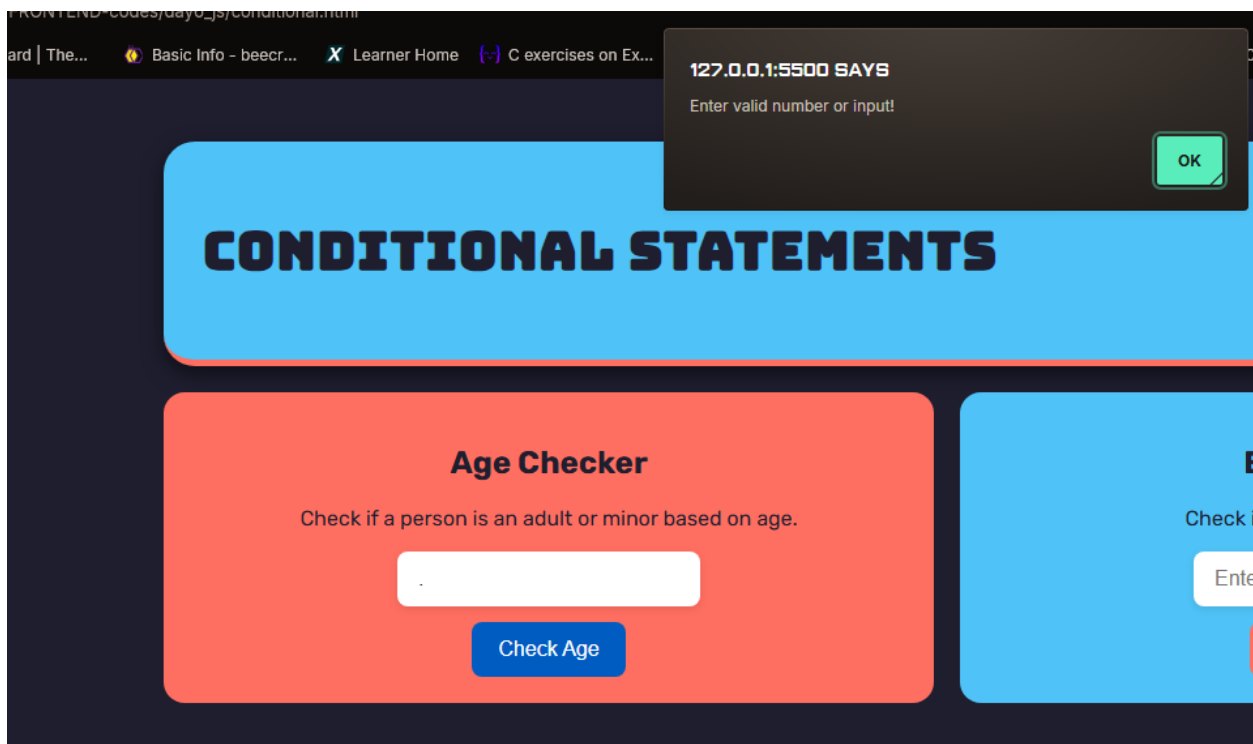
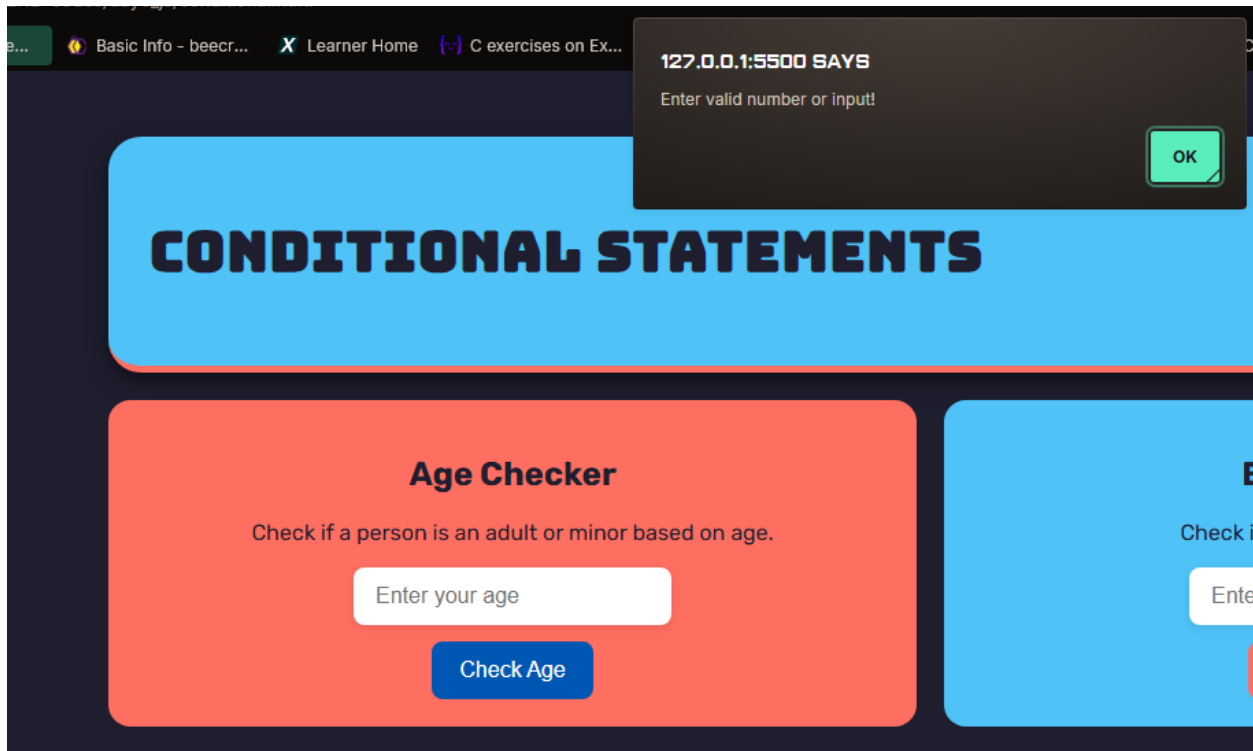
© 2025 John Rich Nicolas. All rights reserved.

127.0.0.1:5500 SAYS  
The number is even.  
OK

STATEMENTS Home Prime Lists

Even or Odd?  
Check if a number is even or odd.  
20  
Check Number

## NULL INPUT





PRIME NUMBERS

PRIME NUMBERS

Home Prime Lists

Show Prime Grid

© 2025 John Rich Nicolas. All rights reserved.

PRIME NUMBERS

Home Prime Lists

Show Prime Grid

© 2025 John Rich Nicolas. All rights reserved.

2	3	5	7
11	13	17	19
23	29	31	37
41	43	47	53

LISTS

LISTS OF NUMBERS

Home Prime Lists

Multiples of 3

Generates a list of the first 10 multiples of 3.

Generate Multiples

Divisible by 5

Shows the last 10 numbers divisible by 5 between 1-100.

Generate Divisibles

MULTIPLES OF 3

LISTS OF NUMBERS

Home Prime Lists

Multiples of 3

Generates a list of the first 10 multiples of 3.

Generate Multiples

3  
6  
9  
12  
15  
18  
21  
24  
27  
30

Divisible by 5

Shows the last 10 numbers divisible by 5 between 1-100.

Generate Divisibles

LISTS OF NUMBERS

Home Prime Lists

Multiples of 3

Generates a list of the first 10 multiples of 3.

Generate Multiples

3  
6  
9  
12  
15  
18  
21  
24  
27  
30

Divisible by 5

Shows the last 10 numbers divisible by 5 between 1-100.

Generate Divisibles

100  
95  
90  
85  
80  
75  
70  
65  
60  
55

## REFLECTION

In my code, I used conditional statements mainly in the age checker and even/odd number checker. For the age checker, I used if and else if to group the ages into categories like adult, teenager, child, or invalid input. For the number checker, I used the modulus operator to check if the number is even or odd, and displayed a message based on the result. I also used loops to generate structured outputs like the 4x4 prime number grid and the lists of numbers. For example, I used a for loop to display the first 10 multiples of 3, and another one to show the last 10 numbers divisible by 5 between 1 to 100. For the prime grid, I used a while loop to keep generating prime numbers until I got 16 of them, and displayed them using createElement.

I organized the logic by putting them into functions, so the code is easier to read and manage. For example, the checkAge(), checkEvenOdd(), generateMultiples(), and generatePrimeGrid() functions each have one job, and I just call them when the button is clicked. I also added input validation by checking if the input is empty or not a number using trim() and isNaN(), so the program doesn't crash or behave weirdly when the user types something invalid. While doing this, I learned about NaN (Not a Number) and how useful it is in catching incorrect inputs like when a user types letters instead of numbers. It helped make my code more reliable and user-friendly. I made sure to use meaningful variable names like ageInput, numberInput, and multiplesList so it's clear what they are for. I tried to keep the structure modular and organized, which helped especially when I was testing each part.

For the styling, I used basic CSS with Flexbox and Grid to layout the pages nicely. I used background colors to separate sections and added some spacing, border-radius, and fonts to make the interface look better and more readable. Something new I also learned was how to use appendChild with loops to dynamically generate content. I also found it interesting to work with user inputs and make sure my code can handle unexpected values. Overall, this activity helped me understand JavaScript better and made me appreciate how functions, validation, and a clean structure make everything easier to work with.