

## 1. Emerging Trends

### Unsolved Exercise ?

#### Section A (Objective Type Questions)

- A.** 1. iii      2. ii      3. iv      4. i      5. iii      6. ii
- B.** 1. Reinforcement learning      2. 2007      3. File management      4. Edge computing  
5. Spam email detection      6. Navigation and decision-making
- C.** 1. False      2. False      3. True      4. False      5. False      6. False

#### Section B (Subjective Type Questions)

- A.** 1. Spreading rumours on social media can lead to social unrest, hostility, and even riots. False information spreads rapidly and can create divisions and conflicts.  
2. Windows, macOS, Linux, Device Drivers.  
3. AI is the simulation of human intelligence in machines programmed to think and learn like humans. ML is a subset of AI that allows machines to learn from data and improve over time.  
4. Cloud computing reduces upfront infrastructure costs and provides pay-as-you-go services, making it cost-efficient.  
5. IoT connects everyday consumer devices, while IIoT connects industrial machines and systems for critical data and high reliability.  
6. Spreading rumours on social media can lead to social unrest, hostility, and even riots. False information spreads rapidly and can create divisions and conflicts.
- B.** 1. Demand for automation, increase in big data usage, advancements in cloud computing, AI and ML integration, government projects for smart cities and 5G technologies.  
2. Edge computing processes data locally, reducing latency. Cloud computing processes data in central servers. Edge computing is faster and reduces bandwidth usage, suitable for real-time applications.

3. Google Drive: Offers 15 GB free storage, integrated with Google services. Dropbox: Offers file synchronisation and sharing, with strong third-party app integrations.
4. Healthcare (IBM Watson), Autonomous Vehicles (Tesla), NLP (Google Assistant), Robotics (Amazon robots).
5. ML assists in diagnosing diseases through medical imaging, predicts patient outcomes, and supports personalised treatment recommendations.
6. Data Collection, Data Preparation, Model Selection, Model Training, Model Evaluation, Parameter Tuning, Prediction are the 7 basic steps of machine learning.

**C. Competency-based/Application-based questions:**

1. Komal should not connect with strangers, protect your privacy, be careful about what she shares.
2. Yes, it can be considered cyberbullying if the post is meant to mock or embarrass you without your consent. It violates privacy and can cause emotional harm.

**D.** 1. c                      2. d

**E.** 1. c                      2. a

## 2. JavaScript Part 2

### Unsolved Exercise ?

#### Section A (Objective Type Questions)

**A.** 1. iii      2. ii      3. ii      4. ii      5. ii      6. i      7. iii      8. iii  
9. iv      10. iii

**B.** 1. typeof                      2. Comments                      3. parameter                      4. shift()  
5. find and replace      6. toLowerCase()                      7. mouseover                      8. Arguments  
9. prompt()                      10. parseInt()

**C.** 1. True      2. False                      3. True                      4. True                      5. False                      6. False  
7. False      8. True



## Section B (Subjective Type Questions)

- A. 1. The pop() method removes the last element in an array and returns it. This modifies the original array, reducing its length by one.

Example:

```
<!DOCTYPE HTML>
<HTML>
<HEAD>
<TITLE>Pop Array Method</TITLE>
</HEAD>
<BODY>
<H2>The pop() Method</H2>
<P>Original Array: </P>
<P id="demo1"></P>
<P>Pop Element: </P>
<P id="demo2"></P>
<P>Array after using Pop method: </P>
<P id="demo3"></P>
<SCRIPT>
const Fruits = ['Apple', 'Mango', 'Banana', 'Papaya'];
document.getElementById("demo1").innerHTML = Fruits;
document.getElementById("demo2").innerHTML = Fruits.pop();
document.getElementById("demo3").innerHTML = Fruits;
</SCRIPT>
</BODY>
</HTML>
```

2. The way that JavaScript interacts with HTML is through events that happen whenever a user or browser modifies a page. An event is defined as a change in an object's state. There are a number of events in HTML that show when a user or browser performs a certain action. The act of responding to events is known as event handling. As a result, JavaScript uses event handlers to handle HTML events. Events are handled by attaching event listeners to HTML elements, which execute a function when the event occurs.

Example:

```
<!DOCTYPE HTML>
<HTML>
```



```

<HEAD>
<TITLE>JavaScript onclick</TITLE>
</HEAD>
<SCRIPT>
function showMessage() {
alert("Button clicked!");
}
</SCRIPT>
</HEAD>
<BODY>
<button onclick="showMessage()">Click Me</button>
</BODY>

```

3. The `math.round()` rounds a number to the nearest integer. The `round()` method rounds a number to the nearest integer, rounding up for decimals  $\geq 0.5$  and down for decimals  $< 0.5$ .

Example:

```

document.write(Math.round(3.68));
document.write(", " + Math.round(5.30));
document.write(", " + Math.round(-8.78));

```

4.	Push()	Unshift ()
	The <code>push()</code> method adds one or more new elements to the end of an array.	The <code>unshift()</code> method adds a new element to an array (at the beginning) and shifts all other elements to a higher index.
	It returns the new length of the array after the element is added.	It returns the new length of the array.
	<b>Example:</b> let arr1 = [1, 2, 3]; arr1.push(4);	<b>Example:</b> let arr1 = [1, 2, 3]; arr1.unshift(0);

5. i. The `math.round()` rounds a number to the nearest integer. The `round()` method rounds a number to the nearest integer, rounding up for decimals  $\geq 0.5$  and down for decimals  $< 0.5$ .

Example:

```

document.write(Math.round(3.68));
document.write(", " + Math.round(5.30));
document.write(", " + Math.round(-8.78));

```



ii. **Math.ceil(x)** The **Math.ceil(x)** method rounds a number up to the nearest integer.

Example:

```
document.write(Math.ceil(4.2));
```

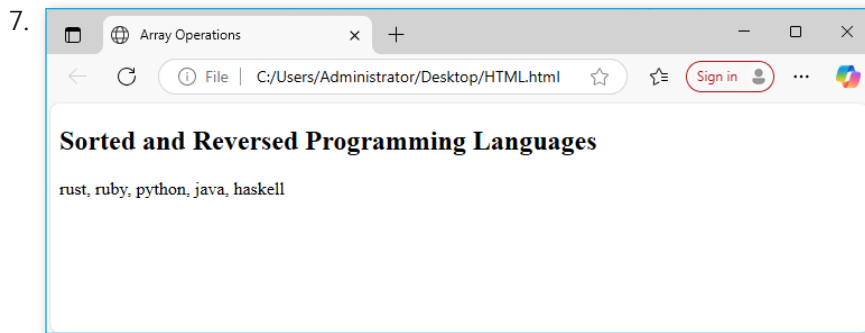
6. • **onmouseover**: Triggered when the mouse cursor approaches over the element

```
<!DOCTYPE HTML>
<HTML>
<HEAD>
<TITLE>JavaScript onmouseover</TITLE>
</HEAD>
<SCRIPT>
function changeColor(element) {
element.style.color = "red";
}
</SCRIPT>
</HEAD>
<BODY>
<p onmouseover="changeColor(this)">Hover over this text</p>
</BODY>
</HTML>
```

- **onmouseout**: Triggered when the mouse cursor exits an element

```
<!DOCTYPE HTML>
<HTML>
<HEAD>
<TITLE>JavaScript onmouseout</TITLE>
</HEAD>
<SCRIPT>
function resetColor(element) {
element.style.color = "black";
}
</SCRIPT>
</HEAD>
<BODY>
<p onmouseout="resetColor(this)" style="color: red;">Move mouse away
from this text</p>
</BODY>
</HTML>
```





8. Far: 37 °C                      Cel: 89.6 °F

**B.** 1. i. The substring() Method

The substring() method extracts characters between two specified indices and returns a new string.

Example:

```
<!DOCTYPE HTML>
<HTML>
<HEAD>
<TITLE>substring() Method</TITLE>
</HEAD>
<BODY>
<H2>The substring() Method</H2>
<P>The substring() method extracts part of a string.</P>
<P id="demo1"></P>
<SCRIPT>
var text = "Hello, JavaScript!";
document.getElementById("demo1").innerHTML = text.substring
(7, 17);
</SCRIPT>
</BODY>
</HTML>
```

ii. The concat() Method

The concat() method joins two or more strings together.

Example:

```
<!DOCTYPE HTML>
<HTML>
```



```

<HEAD>
<TITLE>concat() Method</TITLE>
</HEAD>
<BODY>
<H2>The concat() Method</H2>
<P>The concat() method joins two or more strings.</P>
<P id="demo1"></P>
<SCRIPT>
var text1 = "Hello";
var text2 = "World";
document.getElementById("demo1").innerHTML = text1.concat(" ",
text2);
</SCRIPT>
</BODY>
</HTML>

```

### iii. The Confirm () function

The Confirm() function which prompts the user to press a button. Depending on the button pressed, the respective text is displayed in the body of the webpage.

Example:

```

<!DOCTYPE HTML>
<HTML>
<HEAD>
<TITLE> confirm() Method</TITLE>
</HEAD>
<BODY>
<H2>The confirm() Method</H2>
<p>Click the button to display a confirm box.</p>
<button onclick="myFunction()">Try it</button>
<p id="demo"></p>
<SCRIPT>
function myFunction() {
    let text;
    if (confirm("Press a button!") == true) {
        text = "You pressed OK!";
    }
}

```



```

    } else {
        text = "You canceled!";
    }
    document.getElementById("demo").innerHTML = text;
}
</SCRIPT>
</BODY>
</HTML>

```

#### iv. The shift() Method

The shift() method removes the first element from an array and returns it. This method shifts all other elements in the array to a lower index, effectively reducing the length of the array. The shift() method is similar to the pop() method but here the element removed is the first one instead of the last and hence it is slower. It also returns the removed items.

Example:

```

<!DOCTYPE HTML>
<HTML>
<HEAD>
<TITLE>shift Array Method</TITLE>
</HEAD>
<BODY>
<H2>The shift() Method</H2>
<P>Original Array: </P>
<P id="demo1"></P>
<P>Shift element: </P>
<P id="demo2"></P>
<P>Array after using shift method: </P>
<P id="demo3"></P>
<SCRIPT>
const Fruits = ['Apple', 'Mango', 'Banana', 'Papaya'];
document.getElementById("demo1").innerHTML = Fruits;
document.getElementById("demo2").innerHTML = Fruits.shift();
document.getElementById("demo3").innerHTML = Fruits;
</SCRIPT>
</BODY>
</HTML>

```





2. A block of code created specifically to carry out a specific task is referred to as a JavaScript function. If something calls a JavaScript function, it will be executed (a function call).

The function keyword, a name, and parentheses define a JavaScript function.

```
function displayBiggestRobot() {  
    alert("The biggest robot is Mononofu");0  
}
```

3. The output of the given program is as follows:

ByeBye

4. 

```
function countVowels(str) {  
    let count = 0;  
    const vowels = 'aeiouAEIOU';  
    for (const char of str) {  
        if (vowels.includes(char)) {  
            count++;  
        }  
    }  
    return count;  
}  
  
const userInput = prompt("Please enter a string:");  
const vowelCount = countVowels(userInput);  
alert(`There are ${vowelCount} vowel${vowelCount !== 1 ? 's' : ''}  
in your string.`);
```
5. 

```
function findSecondMax(a, b, c, d, e) {  
    const nums = [a, b, c, d, e];  
    const uniqueNums = Array.from(new Set(nums));  
    uniqueNums.sort((x, y) => y - x);  
    return uniqueNums[1]; // undefined if fewer than 2 distinct  
values  
}  
  
// Prompt the user for five numbers and display the result  
function getUserInputAndShowSecondMax() {  
    const inputs = [];  
    for (let i = 1; i <= 5; i++) {  
        const value = prompt(`Enter number ${i}:`);
```



```

const num = parseFloat(value);
if (isNaN(num)) {
    alert('That is not a valid number. Please reload the page
    and try again.');
```

return;

```

}
inputs.push(num);
}
const secondMax = findSecondMax(...inputs);
if (secondMax === undefined) {
    alert('Not enough distinct numbers provided.');
```

} else {

```

    alert('The second largest number is ${secondMax}.');
}
}
// Run as soon as the page loads
window.onload = getUserInputAndShowSecondMax;
```

## 6. A function without a name, usually assigned to a variable.

The syntax to create anonymous function is as follows:

```

var variableName = function() {
// code block
};
variableName();
```

**Example:**

```

<!DOCTYPE HTML>
<HTML>
<HEAD>
<TITLE>Anonymous Function in JavaScript</TITLE>
</HEAD>
<BODY>
<SCRIPT>
var displayMessage = function() {
document.write("This is an Anonymous Function Example!");
};
```



```
displayMessage();
</SCRIPT>
</BODY>
```

## 7. Key characteristics of anonymous functions in JavaScript:

i. No Identifier: They're defined without a name.

```
// anonymous
function(a, b) { return a + b; }
```

ii. First-Class Citizens: You can treat them like any other value:

- Assign to variables
- Pass as arguments to other functions
- Return from functions

iii. Closures: They capture ("close over") any variables in the surrounding scope, so they can access those even after that scope has exited.

iv. Not Hoisted: Unlike named function declarations, anonymous function expressions aren't hoisted. You must define them before you use them.

v. Common for Callbacks and IIFEs

- Callbacks: e.g. in setTimeout, map, event handlers
- IIFE (Immediately Invoked Function Expression): wraps code in its own scope immediately

vi. Name Property (ES2015+): Modern engines often infer a name property even for anonymous functions when you assign them to variables:

```
let fn = function(a) { return a * 2; };
console.log(fn.name); // "fn"
```

## 8. <!DOCTYPE html>

```
<html>
<head>
<title>Anagram Checker</title>
</head>
<body>
<h1>Check if two strings are anagrams</h1>
<label for="string1">Enter first string:</label>
<input type="text" id="string1"><br><br>
<label for="string2">Enter second string:</label>
<input type="text" id="string2"><br><br>
```



```

<button onclick="getAndCheckAnagram()">Check Anagram</button>
<p id="result"></p>
<script>
    function areAnagrams(str1, str2) {
        // Remove non-alphanumeric characters and convert to lowercase
        // for accurate comparison
        const cleanStr1 = str1.toLowerCase().replace(/[^a-z0-9]/g, '');
        const cleanStr2 = str2.toLowerCase().replace(/[^a-z0-9]/g, '');
        // If the lengths are different, they cannot be anagrams
        if (cleanStr1.length !== cleanStr2.length) {
            return false;
        }
        // Create character maps for both strings
        const charMap1 = {};
        const charMap2 = {};
        for (let char of cleanStr1) {
            charMap1[char] = (charMap1[char] || 0) + 1;
        }
        for (let char of cleanStr2) {
            charMap2[char] = (charMap2[char] || 0) + 1;
        }
        // Compare the character maps
        for (let char in charMap1) {
            if (charMap1[char] !== charMap2[char]) {
                return false;
            }
        }
        return true;
    }
    function getAndCheckAnagram() {
        const string1 = document.getElementById("string1").value;
        const string2 = document.getElementById("string2").value;
        const resultElement = document.getElementById("result");
    }

```



```

    if (string1 && string2) {
        const isAnagram = areAnagrams(string1, string2);
        if (isAnagram) {
            resultElement.textContent = `"${string1}" and "${string2}"
are anagrams.`;
        } else {
            resultElement.textContent = `"${string1}" and "${string2}"
are not anagrams.`;
        }
    } else {
        resultElement.textContent = "Please enter both strings.";
    }
}
</script>
</body>
</html>

```

### C. Competency-based/Application-based questions:

1. `const fruits = ["Apple", "Mangoes", "Orange", "Banana"];`  
`const veg = ["Potato", "Brinjal", "Gourd"];`
  - i. Add "Guava" to the end of the fruits array:  
`fruits.push("Guava");`  
`console.log("Fruits after adding Guava:", fruits);`
  - ii. Remove the first element from the fruits array:  
`const removedFruit = fruits.shift();`  
`console.log("Fruits after removing the first element:", fruits);`  
`console.log("Removed fruit:", removedFruit);`
  - iii. Display the number of elements in the fruits array:  
`const numberOfFruits = fruits.length;`  
`console.log("Number of elements in the fruits array:", numberOfFruits);`
  - iv. Add the "veg" array to the "fruits" array:  
`fruits.push(...veg);`  
`console.log("Fruits after adding the veg array:", fruits);`



```

2. function stringsinjava(str1, str2) {
    // 1. Convert both strings to uppercase.
    const upperStr1 = str1.toUpperCase();
    const upperStr2 = str2.toUpperCase();
    console.log("Uppercase String 1:", upperStr1);
    console.log("Uppercase String 2:", upperStr2);
    // 2. Search for string1 (case-insensitive now) in string2 and display
    if found.
    if (upperStr2.includes(upperStr1)) {
        console.log(`"${str1}" found in "${str2}" (case-insensitive
            check: "${upperStr1}" in "${upperStr2}").`);
    } else {
        console.log(`"${str1}" not found in "${str2}" (case-insensitive
            check: "${upperStr1}" in "${upperStr2}").`);
    }
    // 3. Replace all occurrences of letter 'A' (case-insensitive) with
    '*' in string1.
    const replacedStr1 = str1.toUpperCase().split('A').join('*');
    console.log(`String 1 after replacing 'A' with '*':`, replacedStr1);
    // 4. Display the first character of string2.
    if (str2.length > 0) {
        console.log("First character of String 2:", str2.charAt(0));
    } else {
        console.log("String 2 is empty, cannot display the first
            character.");
    }
}

// Example usage of the function:
const string1Input = "apple";
const string2Input = "Big Apple Pie";
stringsinjava(string1Input, string2Input);
console.log("\n--- Another Example ---");
const anotherString1 = "banana";
const anotherString2 = "Orange and Banana";
stringsinjava(anotherString1, anotherString2);

```



```
console.log("\n--- Example with no 'a' in string1 ---");
const noAString1 = "grape";
const someString2 = "Some Text";
stringsinjava(noAString1, someString2);
console.log("\n--- Example with empty string2 ---");
const textString1 = "hello";
const emptyString2 = "";
stringsinjava(textString1, emptyString2);
```

- D.** 1. a. Both A and R are true and R is the correct explanation of A.  
2. d. A is false but R is true.
- E.** 1. c. Statement 1 is True, Statement 2 is False.  
2. c. Statement 1 is True, Statement 2 is False.

### 3. Graphic Designing Using Canva and Adobe Express

#### Unsolved Exercise ?

##### Section A (Objective Type Questions)

- A.** 1. iv      2. i      3. i      4. iii      5. iii      6. i      7. ii      8. ii  
9. iii      10. iii
- B.** 1. visual      2. Balance      3. Download      4. text, colours, icons  
5. Canva      6. audio
- C.** 1. True      2. False      3. True      4. True      5. True      6. False

##### Section B (Subjective Type Questions)

- A.** 1. Two key features of Adobe Express:
- Quick content creation with templates.
  - AI-powered photo enhancement and video editing tools.
2. Advantage of using templates:
- Saves time and ensures professional-quality designs.
3. Significance of typography in design:
- Enhances readability and conveys the tone or emotion of the design.



4. How Canva's AI feature improves design creation:
  - It automates design adjustments for better visual appeal.
5. Two file formats supported by Canva:
  - PNG and MP4.
6. How to export a video from Adobe Express:
  - Click on the "Download" option and choose the desired video format.

**B. 1. Steps to create a blank presentation in Canva:**

Step 1: Log in to your Canva account.

Step 2: On the homepage, click on the "Create a design" button.

Step 3: Select "Presentation" from the available design types.

Step 4: Choose "Blank Presentation" instead of pre-made templates.

Step 5: A new editor window will open with blank slides. You can start adding text, images, videos, and elements to create your presentation.

Step 6: Use Canva's design tools and features to customise your slides.

Step 7: Once done, click "Download" or use "Present" to share your presentation.

**2. Explain the concept of AI integration in Canva and its benefits:**

Canva integrates Artificial Intelligence (AI) to simplify the design process for users. AI tools in Canva help by automating layout adjustments, suggesting design elements, and generating text and images. Features like Magic Resize, Background Remover, and Design Suggestions are AI-driven, enabling users to create professional designs quickly without advanced skills.

Benefits include:

- Saves time by automating repetitive tasks.
- Enhances creativity with smart recommendations.
- Improves design accuracy and visual appeal.
- Makes designing accessible even for beginners.

**3. Steps to design using a template in Adobe Express:**

Step 1: Open Adobe Express and sign in to your account.

Step 2: Click on "Create new" and select the desired project type (e.g., poster, logo, flyer).

Step 3: Browse through the available templates or use the search bar to find a suitable one.

Step 4: Select a template to open it in the editor.

Step 5: Customise the template by editing text, changing colours, and adding images or icons.

Step 6: Use design tools to enhance your project (apply filters, adjust layout, etc.).

Step 7: Once satisfied, click on "Download" or "Share" to export your design.





4. How do you add audio tracks to a video in Adobe Express?

Step 1: Open Adobe Express and start a new video project.

Step 2: Upload your video clips to the timeline.

Step 3: Click on the "Audio" option in the toolbar.

Step 4: Choose from available music tracks in Adobe Express or upload your own audio file.

Step 5: Drag and drop the selected audio track onto the timeline below your video.

Step 6: Adjust the audio timing and volume as needed.

Step 7: Preview your video and make final adjustments.

Step 8: Click "Download" to export your video with the added audio track.

**C. Competency-based/Application-based questions:**

1. Use Canva or Adobe Express to quickly create the invitation cards and use the "Share" option to instantly send the design link to your teachers for review and approval.
2. Create the save the date video using Canva or Adobe Express, then generate a shareable link using the "Share" feature and send it to family and friends.

- D.** 1. a. Both A and R are true and R is the correct explanation of A.  
2. a. Both A and R are true and R is the correct explanation of A.

- E.** 1. b. Statement 1 is False, Statement 2 is False  
2. b. Statement 1 is False, Statement 2 is False

## 4. Cyber Safety and Security

### Unsolved Exercise ?

#### Section A (Objective Type Questions)

- A.** 1. iii      2. iv      3. ii      4. iv      5. iii
- B.** 1. Identity theft      2. Email spoofing      3. Cyberstalking  
4. Copyright      5. Plagiarism
- C.** 1. False      2. True      3. True      4. False      5. True

#### Section B (Subjective Type Questions)

- A.** 1. Responsible online behaviour ensures safe, respectful, and secure internet use. Cyber laws help by providing legal protection, penalising wrongdoers, and promoting accountability.



2. Cyberbullying involves sharing harmful or private information to embarrass someone online, while cyberstalking involves continuous harassment by tracking personal information or excessive communication.
  3.
    - a. Patent - Innovative machine
    - b. Copyright - Book or software code
    - c. Trademark - Logo or brand name
  4.
    - Use strong passwords
    - Install antivirus software
  5. Copyright grants the owner exclusive rights to reproduce, distribute, publicly perform, display, and adapt their work.
- B.**
1.
    - Use strong, unique passwords and change them regularly.
    - Install and update antivirus and anti-malware software.
    - Enable two-factor authentication (2FA) for online accounts.
    - Avoid clicking on suspicious links or downloading attachments from unknown sources.
    - Keep software, apps, and operating systems updated.
    - Regularly back up important data to a secure location.
    - Use secure Wi-Fi networks and avoid using public Wi-Fi for sensitive transactions.
    - Be cautious while sharing personal information online.
  2.
    - Patent: Protects new inventions, processes, or technological solutions. It gives the inventor exclusive rights to use, make, or sell the invention for a certain period.
    - Copyright: Protects original creative works such as books, music, art, and software code. It gives the creator the right to reproduce, distribute, and display their work.
    - Key difference: A patent protects inventions and processes, while copyright protects creative expressions.
  3.
    - DoS stands for Denial of Service attack.
    - The attacker floods a network or website with excessive traffic to overwhelm the system.
    - It causes legitimate users to be unable to access services or information.
    - It disrupts business operations and can lead to financial losses.
    - DoS attacks often target websites, servers, or online services.
    - Attackers use botnets or malicious scripts to automate the traffic overload.
  4.
    - Email Spoofing is a cyber-attack where the attacker forges the sender's address to make an email appear as if it is from a trusted source.
    - The goal is usually to trick recipients into revealing sensitive information or clicking malicious links.
    - It is commonly used in phishing attacks to deceive users.



5.
  - A Denial of Service (DoS) attack is an attempt by cybercriminals to make a network service unavailable to its intended users by overwhelming it with a flood of unnecessary traffic.
  - It exhausts the resources of the server, causing slowdowns or complete shutdowns.
  - DoS attacks disrupt normal functioning, leading to downtime and potential financial and reputational damage.

**C. Competency-based/Application-based questions:**

1. Red flags include unknown sender, request for personal info, and urgency. Verify by checking official sources and not sharing personal details.
2. Do not share the meme and explain to friends it's disrespectful. Promote kindness online and avoid supporting cyberbullying.

**D.** 1. d. A is false but R is true.

2. a. Both A and R are true and R is the correct explanation of A.

**E.** 1. d. Statement 1 is False, Statement 2 is True.

2. d. Statement 1 is False, Statement 2 is True.

