WEB TECHNOLOGY AND FRAMEWORKS

UNIT – 1

1.ANS

```
        <nav>
            <h1>Food Delivery</h1>
            <ul>
                <li><a href="#">Home</a></li>
                <li><a href="#">Menu</a></li>
                <li><a href="#">Orders</a></li>
                <li><a href="#">Contact</a></li>
            </ul>
        </nav>
    </header>

    <div>
        <!-- (ii) Add the main content area for food listings or featured
        <main>
            <section>
                <h2>Featured Dishes</h2>
                <p>Discover the best food available for delivery!</p>
            </section>
            <section>
                <h2>Popular Restaurants</h2>
                <p>Check out top-rated restaurants near you.</p>
            </section>
        </main>
```

2.ANS:

```
<table style="width:100%">
    <tr>
        <th>Firstname</th>
        <td>Jill</td>
        <td>Eve</td>
    </tr>
    <tr>
        <th>Lastname</th>
        <td>Smith</td>
        <td>Jackson</td>
    </tr>
    <tr>
        <th>Age</th>
        <td>50</td>
        <td>94</td>
    </tr>
</table>

</body>
</html>
```

3.ANS:

Precise 2-Mark Answers for Full Score (100 Marks)

(a) Impact of Incorrect DNS Configuration:

Incorrect DNS settings can make the website inaccessible, increase latency, misroute traffic, and lead to revenue loss.

(b) Load Balancer Mechanism:

A load balancer distributes user requests across multiple servers, prevents overload, ensures failover, and improves performance.

(c) Contribution of Caching:

Caching reduces server load, speeds up response time, minimizes database queries, and enhances user experience.

5.ANS:

Correct Answer: (a) Both A and R are true, and R correctly explains A.

Justification:

- The **Assertion (A)** states that a **three-tier architecture** is widely used due to its structured and modular approach, which is **true**.
- The **Reason (R)** correctly describes the **three layers** in a three-tier architecture:
  1. **Presentation Layer** – Manages user interactions and UI.
  2. **Business Logic Layer** – Handles core processing and business rules.
  3. **Data Access Layer** – Manages database storage and retrieval.
- Since the reason **accurately explains** why three-tier architecture is modular and organized, **option (a) is correct**. ✅

6.ANS:

```
<!DOCTYPE html>
<html>
<head>
  <style>
    #container {
      background-color: yellow;
    }
  </style>
</head>
<body>

<div id="container">Content</div>

<script>
  // This script is not necessary to change the background color
  // as it's already defined in the style section
  // Color = "yellow";
</script>

</body>
</html>
```

UNIT – 2

1.ANS:

a) Benefits of Cross-Platform Development for a Fitness App

1. **Cost-Efficiency**: Only one codebase is needed, reducing development and maintenance costs.
2. **Faster Development**: Features can be implemented quickly and simultaneously for iOS and Android.
3. **Uniform Experience**: Ensures a consistent look, feel, and behavior on both platforms.
4. **Easy Updates**: Bug fixes and new features can be deployed to both platforms at the same time.
5. **Strong Community and Libraries**: Tools like Flutter's extensive libraries accelerate development.

b) Flutter's High Performance and Native-Like Experience

1. **Native Compilation**: Flutter compiles to native ARM or x64 code, reducing latency.
2. **Custom Rendering Engine**: Skia ensures smooth animations and UI rendering.
3. **Efficient Widgets**: Flutter's widgets are highly optimized for performance.
4. **Asynchronous Handling**: Dart handles background tasks seamlessly without UI lag.
5. **Hot Reload**: Speeds up development and reduces debugging time, ensuring efficient performance.

c) Maintaining Consistency with Flutter

1. **Single Codebase**: Facilitates consistent functionality and design across platforms.
2. **Platform-Specific Widgets**: Material Design for Android and Cupertino for iOS ensures familiarity.
3. **Centralized State Management**: Libraries like Riverpod or Redux ensure uniform behavior.
4. **Custom Widgets**: Enable creating unique designs that look consistent across platforms.
5. **Integrated Testing**: Flutter allows testing for both platforms from the same codebase.

2.ANS:

```javascript
function binarySearch(arr, target) {
  let left = 0;
  let right = arr.length - 1;

  while (left <= right) {
    let mid = Math.floor((left + right) / 2);
    if (arr[mid] === target) {
      return mid;
    } else if (arr[mid] < target) {
      left = mid + 1;
    } else {
      right = mid - 1;
    }
  }
  return -1;
}

// Test Cases
console.log(binarySearch([1, 5, 6, 8, 10, 12, 14], 12)); // Output: 5
console.log(binarySearch([1, 13, 5, 7], 4)); // Output: -1
```

3.ANS:

JavaScript: Find Two Sum Indices

```javascript
function twoSum(nums, target) {
  let numMap = new Map();

  for (let i = 0; i < nums.length; i++) {
    let complement = target - nums[i];
    if (numMap.has(complement)) {
      return [numMap.get(complement), i];
    }
    numMap.set(nums[i], i);
  }
  return [];
}

// Test Case
console.log(twoSum([3, 8, 12, 17], 20)); // Output: [1, 2]
```

4.ANS:

javascript

Copy Edit

```javascript
function maxProfit(prices) {
  let minPrice = Infinity; // Initialize minimum price to a very high value
  let maxProfit = 0; // Initialize maximum profit to 0

  for (let i = 0; i < prices.length; i++) {
    if (prices[i] < minPrice) {
      minPrice = prices[i]; // Update minPrice if a lower price is found
    }
    let potentialProfit = prices[i] - minPrice; // Calculate potential profit
    if (potentialProfit > maxProfit) {
      maxProfit = potentialProfit; // Update maxProfit if it's greater
    }
  }
  return maxProfit; // Return the maximum profit found
}

// Test Case
const prices = [10, 2, 8, 1, 9];
console.log(maxProfit(prices)); // Output: 8
```

5.ANS:

HTML Fix

Corrected code:

html

Copy Edit

```html
<!DOCTYPE html>
<html>
<body>
  <p id="message">Old Message</p>
  <button onclick="changeText()">Click Me</button>

  <script>
    function changeText() {
      document.getElementById("message").innerHTML = "New Message";
    }
  </script>
</body>
</html>
```

6.ANS:

```
outer();
```

i) Scope of `y`:

- `y` is local to `inner()` and not accessible outside of it.

ii) Predictions:

1. `console.log(x + y)` inside `inner()`: Logs **15**. `x` is in `outer()`'s scope, and `y` is in `inner()`'s local scope.
2. `console.log(x + y)` outside `inner()`: Throws a **ReferenceError** because `y` is not defined in `outer()`.

UNIT – 3

1.ANS:

Here's the correct matching of the HTTP Status Codes, their Information, and the Error Handling Methods based on the provided image:

1. **A1. 400 - B2. Bad Request - C2. Validate client input; provide an error message to the user indicating the problem with the request.**
2. **A2. 403 - B4. The server understood the request, but the client does not have permission - C1. Inform the user that they are not authorized to perform the action, and suggest obtaining permissions.**
3. **A3. 401 - B5. Prompt the user to log in or provide valid authentication credentials - C4. The request requires user authentication.**
4. **A4. 201 - B3. The request has been fulfilled, and a new resource has been created - C3. Notify the user of the successful creation, and possibly display the new resource's details.**
5. **A5. 503 - B1. The server is temporarily unavailable due to maintenance or overload - C5. Inform the user that the server is temporarily unavailable and suggest retrying later.**

2.ANS:

Here are the correct answers for the statements provided:

a. **Represents a variety of resources, including HTML pages, images, videos, etc.**
Answer: URI
Explanation: A URI is a broader concept that represents resources of all types.

b. **Is always absolute but never relative.**
Answer: FQDN
Explanation: An FQDN specifies the absolute domain name, always including the root domain.

c. **Can be used for naming schemas and identification.**
Answer: URN
Explanation: A URN is used as a unique identifier for resources, focusing on naming rather than locating.

d. **Specifies the exact location of a resource on the internet.**
Answer: URL
Explanation: A URL provides the specific location of a resource using schemes like `http`, `ftp`, etc.

e. **Contains the protocol or scheme used to access the resource.**
Answer: URL
Explanation: A URL specifies the scheme or protocol (e.g., `http`, `https`) used to access the

Message ChatGPT

f. **May include parameters or query strings to define specific actions or filters.**
Answer: URL
Explanation: URLs can include parameters or query strings for defining actions or filtering data.

g. **Can be used for a wide range of applications beyond the web, such as databases and file systems.**
Answer: URI
Explanation: URIs are used in multiple contexts, including databases, file systems, and web applications.

h. **Provides both the location and resource identifier for accessing a web resource.**
Answer: URL
Explanation: A URL specifies both the location and the identifier, combining the resource's address and access method.

3.ANS:

Matches:

A1. PUT

- **B2:** Sends data to the server to be processed
- **C4:** Updates an existing resource with new data

A2. GET

- **B1:** Retrieves data from the server
- **C1:** Does not change the resource

A3. PATCH

- **B4:** Partially modifies an existing resource
- **C3:** Updates the resource with the provided data

A4. DELETE

- **B3:** Deletes a specified resource
- **C2:** Removes the resource from the server

4.ANS:

Answer:

Question 4: Assign Responsibilities for Model, View, and Controller

Model:

- B) Data Storage and Retrieval
- H) Data Structure Representation

View:

- C) Interface Design
- D) Displaying Information

Controller:

- A) Core Functionality
- E) Middle Layer Interaction
- F) Handling User Input
- G) Managing Requests