Alright 👍 below are **very long, clear, exam-oriented answers** for **all 11 questions**, written exactly in the **LPU / Web Technologies** style. You can safely write these in **10–15 mark** answers.

---

1. What's the difference between window and document?

The **window** and **document** objects are both part of the **Browser Object Model (BOM)** and **Document Object Model (DOM)**, but they serve different purposes.

window Object:

- It is the **top-level global object** in the browser.
- Represents the **browser window or tab**.
- All global JavaScript variables and functions become properties of window.
- Controls browser-level features.

Examples:

window.alert("Hello");

window.open("https://example.com");

document Object:

- It is a **child object of window**.
- Represents the **HTML document loaded inside the browser window**.
- Used to access and manipulate HTML elements.

Example:

document.getElementById("demo").innerHTML = "Hello";

Key Differences:

Window	Document
Represents browser window	Represents HTML page
Part of BOM	Part of DOM
Controls browser actions	Controls page content
Parent object	Child of window

2. What are the methods to get one or more elements from the DOM?

DOM provides several methods to **access HTML elements**.

Common DOM methods:

1. **getElementById()**

document.getElementById("id");

Returns a single element.

2. **getElementsByClassName()**

document.getElementsByClassName("class");

Returns a collection of elements.

3. **getElementsByTagName()**

document.getElementsByTagName("p");

Returns elements by tag name.

4. **querySelector()**

document.querySelector(".box");

Returns the first matching element.

5. **querySelectorAll()**

document.querySelectorAll(".box");

Returns all matching elements.

These methods allow JavaScript to dynamically manipulate HTML.

---

3. What is the main function of DOM?

The **main function of the DOM (Document Object Model)** is to provide a **structured representation of an HTML document** and allow JavaScript to interact with it.

Main functions:

- Access HTML elements
- Modify content dynamically
- Change styles
- Handle user events
- Add or remove elements

Example:

document.getElementById("demo").style.color = "red";

DOM makes webpages **interactive and dynamic** without reloading.

---

4. What is the method involved in HTML DOM?

HTML DOM provides various **methods** to manipulate HTML elements.

Important HTML DOM methods:

- getElementById()

- getElementsByClassName()
- getElementsByTagName()
- createElement()
- appendChild()
- removeChild()
- replaceChild()
- setAttribute()
- getAttribute()

Example:

var para = document.createElement("p");

para.innerHTML = "New Paragraph";

document.body.appendChild(para);

These methods allow full control over HTML structure.

---

5. What are the functionalities performed by onload() and onUnload()?

onload():

- Executes when a webpage **finishes loading completely**
- Used to initialize scripts and elements

Example:

<body onload="loadMessage()">

onUnload():

- Executes when the user **leaves the webpage**
- Used to clean up resources or show confirmation messages

Example:

<body onunload="goodbye()">

Functions:

- onload() → Page initialization
- onUnload() → Page exit handling

---

6. What are HTML DOM Events?

HTML DOM Events are **actions performed by users or browsers** that JavaScript can respond to.

Types of Events:

- Mouse events (onclick, onmouseover)
- Keyboard events (onkeydown, onkeyup)
- Form events (onsubmit, onchange)
- Window events (onload, onresize)

Example:

<button onclick="alert('Clicked!')">Click</button>

Events make webpages interactive.

---

7. What are JavaScript Cookies?

Cookies are **small text files** stored in the user's browser by websites.

Uses of Cookies:

- Store user preferences
- Maintain login sessions
- Track user behavior

Example:

document.cookie = "username=John; expires=Fri, 31 Dec 2026";

Characteristics:

- Stored on client-side
- Limited size (~4KB)
- Sent with every HTTP request

---

8. What is Ajax?

AJAX stands for **Asynchronous JavaScript and XML**.

Purpose:

- Communicate with server **without reloading the page**

Technologies used:

- JavaScript
- XML / JSON
- XMLHttpRequest / Fetch API

Example use cases:

- Live search
- Form submission

- Auto-updating content

AJAX improves performance and user experience.

---

9. What is Cookie String?

A **cookie string** is a single string containing all cookies stored for a webpage.

Format:

name=value; name2=value2;

Example:

document.cookie = "user=Krishna";

Access:

var cookies = document.cookie;

Cookie strings are used to read, write, and manage cookies.

---

10. What are the JSON Techniques?

JSON (JavaScript Object Notation) is a lightweight data exchange format.

JSON Techniques:

1. **Parsing JSON**
2. **Stringifying JSON**
3. **Data exchange between server and client**

Methods:

- JSON.parse()
- JSON.stringify()

JSON is language-independent and easy to use.

---

11. How parsing is done in JSON? Explain with example

Parsing in JSON converts a **JSON string into a JavaScript object**.

Method:

JSON.parse()

Example:

var jsonData = '{"name":"Krishna","age":22}';

var obj = JSON.parse(jsonData);

document.write(obj.name);

Explanation:

- JSON string → JavaScript object
- Allows access to properties using dot notation

Parsing is essential for handling server responses.
