**ReactJS Assignment**

**Module – 2 : Advance JavaScript**

**Q 2 - What is JavaScript Output method?**

Ans -

JavaScript has several methods for outputting content to the user interface. Here are the main ones:

1. console.log(): Outputs a message to the web console. Mostly used for debugging purposes.

2. alert(): Displays a dialog box with a message and an OK button. Used to alert users or get their attention.

3. document.write(): Writes HTML expressions or JavaScript code to a document. Useful for testing and generating dynamic content.

4. innerHTML: Allows you to change the content of an HTML element. You can set the content of an element using this property.

5. textContent: Similar to innerHTML, but it sets or returns only the text content of the specified element, without any HTML tags.

Each method has its own use case and is suitable for different scenarios depending on the desired outcome.

**Q 3 - How to used JavaScript Output method?**

Ans :-

Certainly! Here's a summary in bullet points:

* Choose a Method: Decide which output method to use based on your needs (e.g., `console.log()`, `alert()`, `document.write()`, `innerHTML`, `textContent`).
* Write JavaScript: Use the selected method in your JavaScript code. For example:

- `console.log("Hello, world!");`

- `alert("This is an alert!");`

- `document.write("Dynamic content");`

- `document.getElementById("output").innerHTML = "Hello, world!";`

- `document.getElementById("output").textContent = "Hello, world!";`

* Integrate with HTML: If updating HTML content, ensure you target the correct element. For example:

**<div id="output"> </div>**

* Run Your Code: Save your HTML file and open it in a web browser to see the output.

**Q 4 - How to used JavaScript Events to do all examples?**

**Ans –**

Here is a list of some commonly used JavaScript events:

Mouse Events:

click: Occurs when the mouse is clicked on an element.

dblclick: Occurs when the mouse is double-clicked on an element.

mousedown: Occurs when a mouse button is pressed down on an element.

mouseup: Occurs when a mouse button is released over an element.

mousemove: Occurs when the mouse pointer moves over an element.

mouseover: Occurs when the mouse pointer moves onto an element.

mouseout: Occurs when the mouse pointer moves out of an element.

mouseenter: Similar to mouseover, but does not bubble up.

mouseleave: Similar to mouseout, but does not bubble up.

Keyboard Events:

keydown: Occurs when a key is pressed down.

keyup: Occurs when a key is released.

keypress: Occurs when a key is pressed and released.

Form Events:

submit: Occurs when a form is submitted.

reset: Occurs when a form is reset.

change: Occurs when the value of an element changes (for <input>, <select>, and <textarea>).

input: Occurs when the value of an <input> or <textarea> element changes (similar to change, but fires immediately).

Window Events:

load: Occurs when the document and all external resources have finished loading.

unload: Occurs when the document is being unloaded (e.g., when the user navigates away from the page).

resize: Occurs when the browser window is resized.

scroll: Occurs when the document is scrolled.

Focus Events:

focus: Occurs when an element gains focus.

blur: Occurs when an element loses focus.

Media Events:

play: Occurs when the media playback has started.

pause: Occurs when the media playback has been paused.

ended: Occurs when the media playback has reached the end.

Drag Events:

dragstart: Occurs when a drag operation starts.

drag: Occurs when an element is being dragged.

dragend: Occurs when a drag operation ends.

Touch Events:

touchstart: Occurs when a touch point is placed on the touch surface.

touchmove: Occurs when a touch point is moved along the touch surface.

touchend: Occurs when a touch point is removed from the touch surface.

These are just a few examples of JavaScript events. There are many more events available, and you can also create custom events using the CustomEvent constructor.