

**Q1.State what is a cookie? Explain its need.]**

- Cookies are data, stored in small text files, on your computer.
- When a web server has sent a web page to a browser, the connection is shut down, and the server forgets everything about the user.
- Cookies can be used for a variety of purposes, such as remembering a user's preferences, tracking their behavior on a website, and enabling personalized advertising.
- There are two main types of cookies: session cookies and persistent cookies.
- Cookies are saved in name-value pairs like: username = John Doe

Need

1. Cookies are small files created by websites to store information about your visit.
2. They help websites remember who you are and make your experience smoother.
3. Cookies can store your preferences, like language or theme settings.
4. They keep you logged in so you don't need to enter your details each time.
5. Cookies track how long you stay on a website and which pages you visit.
6. In online stores, cookies help save items in your shopping cart as you browse.
7. Overall, they make web browsing more personalized and convenient.

**Q2.Descibe and explain types of cookies . i)Session cookie and ii) persistent cookies.**

**Session cookies:**

Session cookies are also known as temporary cookies which are present as long as the user browser is open. Session cookies are deleted once the browser is closed and the user's session is inactive (time-based). Session cookies are most often used to maintain the user's session ( like authentication ) on the browser.

**Persistent cookies:**

Persistent cookies are also known as permanent/long-term cookies. Persistent cookies can last longer than session cookies. Persistent cookies are stored on the user's device for a specific period ( for less than 6 months ). Persistent cookies are most often used for long-term tracking and remembering user preferences.

Feature	Session Cookie	Persistent Cookie
Duration	Active only during the browser session	Remains on the device even after closing the browser
Expiry	Deleted once the browser is closed	Stored until a set expiration date or manually deleted
Purpose	Tracks temporary information (e.g., shopping cart items)	Stores user preferences (e.g., login details) for future visits
Storage Location	Stored in the browser's memory	Stored on the user's hard drive (or device)
Usage	Ideal for temporary data or session management	Used for long-term data retention like user authentication
Security	Considered safer, as it is deleted after each session	May pose more risk if not properly secured, as they persist over time
Examples	E-commerce cart during a session	"Remember me" login option, language settings

### Q3. Describe or explain create, read, write , update and delete cookies.

#### Create a Cookie / Write a Cookie

Creating or writing a cookie involves storing a small piece of data in the user's browser. This is done using JavaScript by setting a string in the document.cookie object. A cookie typically has a name, a value, and an optional expiration date.

```
document.cookie = "username=JohnDoe; expires=Fri, 30 Dec 2024 12:00:00 UTC; path=/";
```

- **username=JohnDoe:** This defines the name and value of the cookie (username is the name, and JohnDoe is the value).
- **expires=Fri, 30 Dec 2024 12:00:00 UTC:** Sets the expiration date for the cookie. If not set, the cookie becomes a session cookie and is deleted when the browser is closed.
- **path=/:** Specifies the path for which the cookie is valid. / means the cookie is accessible on the entire website

```
<html>
  <head>
    <script language="javascript" type="text/javascript">
      function CreateCookie() {
        var a = document.getElementById('tid1').value;
        document.cookie="Username="+a;
        alert("Cookie Created");
        alert(document.cookie);
      }
    </script>
  </head>
  <body>
    <form name="form1" id="fid">
      Enter Username :
      <input type="text" name="tf1" id="tid1">
      <br><br>
      <input type="button" name="b1" onclick="CreateCookie()" value="Create">
    </form>
  </body>
</html>
```

## Read a cookie

When a cookie is created it can be read by the browser. Reading a cookie is just as simple as writing one, because the value of the document.cookie object is the cookie. So you can use this string whenever you want to access the cookie. The document.cookie string will keep a list of name=value pairs separated by semicolons, where name is the name of a cookie and value is its string value.

```
<html>
<head>
<script type="text/javascript">
function ReadCookie() {
    var allcookies = document.cookie; // Get all cookies as a single string
    var cookiearray = allcookies.split(';'); // Split cookies into array

    // Loop through the cookie array and display each key-value pair
    for (var i = 0; i < cookiearray.length; i++) {
        var cookie = cookiearray[i].split('='); // Split into key and value
        var name = cookie[0].trim(); // Get the cookie name
        var value = cookie[1]; // Get the cookie value
        alert("Key: " + name + " | Value: " + value); // Show key-value pair
    }
}
</script>
</head>
<body>
<form name="myform">
    <p>Click the button to get cookies:</p>
    <input type="button" value="Get Cookie" onclick="ReadCookie()"/>
</form>
</body>
</html>
```

username=JohnDoe; theme=dark; sessionToken=abc123

Key: username | Value: JohnDoe

Key: theme | Value: dark

Key: sessionToken | Value: abc123

## Update a Cookie

Update a cookie, you simply create a new cookie with the same name but with an updated value or expiration date. The browser will overwrite the old cookie with the new one.

// Update the "username" cookie to a new value and reset the expiration date

document.cookie = "username=JaneDoe; expires=Fri, 30 Dec 2024 12:00:00 UTC; path=/";

```
<html>
<head>
<script type="text/javascript">
function updateCookie() {
    // Update the "username" cookie to a new value and set the expiration date
    document.cookie = "username=JaneDoe; expires=Fri, 30 Dec 2024 12:00:00 UTC; path=/";
    alert("Cookie 'username' updated to 'JaneDoe'");
}
</script>
</head>
<body>
<form name="myform">
    <p>Click the button to update the 'username' cookie:</p>
    <input type="button" value="Update Cookie" onclick="updateCookie()"/>
</form>
</body>
</html>
```

## Delete a Cookie

To delete a cookie, you set its expiration date to a past date. When a browser detects that a cookie has expired, it automatically deletes the cookie.

document.cookie = "username=; expires=Thu, 01 Jan 1970 00:00:00 UTC; path=/";

This code deletes the username cookie by setting its value to an empty string and its expiration date to January 1, 1970, which is considered a past date.

```
<html>

<head>

<script type="text/javascript">

function deleteCookie() {

    // Delete the "username" cookie by setting its value to an empty string and expiration
    date to a past date

    document.cookie = "username=; expires=Thu, 01 Jan 1970 00:00:00 UTC; path=/";

    alert("Cookie 'username' has been deleted");

}

</script>

</head>

<body>

<form name="myform">

    <p>Click the button to delete the 'username' cookie:</p>

    <input type="button" value="Delete Cookie" onclick="deleteCookie()"/>

</form>

</body>

</html>
```

### Setting the Expiration Date of Cookie

- You can extend the life of a cookie by setting a new expiration date to the cookie.
- This can be done by setting the 'expires' attribute to a new date and time.
- The new date object is obtained by using '*new Date()*'.
- The Date has following function to get the month of the date to set the month of the date and to format the date string.
  1. **getMonth()** : this function returns the month of the current date object.
  2. **setMonth()** : this function set the month to the current date object
  3. **toUTCString()** : this function formats the date in the standard format to set to the expiration date of the cookie.

## Program 1 : Write a javascript program to create read, update and delete cookies (4marks)

```
<!doctype html>
<html>
<head>
  <title>Cookie Operations</title>
</head>
<body>
  <h2>Performing cookie operations</h2>
  <form name="my">
    <input type="text" id="cookie" placeholder="Enter cookie value">
    <button type="button" onclick="CreateCookie()">Create Cookie</button>
    <button type="button" onclick="ReadCookie()">Read Cookie</button>
    <button type="button" onclick="UpdateCookie()">Update Cookie</button>
    <button type="button" onclick="DeleteCookie()">Delete Cookie</button>
  </form>

  <script type="text/javascript">
    function CreateCookie() {
      var a = document.getElementById("cookie").value;
      document.cookie = "username=" + a + "; expires=Sun, 20 Oct 2024 12:00:00
UTC; path=/";
      alert("Cookie 'username' created with value: " + a);
    }

    function ReadCookie()
    {
      var rcookie = document.cookie;
      alert("Current Cookies: " + rcookie);
    }

    function UpdateCookie() {
      var a = prompt("Enter updated cookie value:");
      document.cookie = "username=" + a + "; expires=Sun, 20 Oct 2024 12:00:00
UTC; path=/";
      alert("Cookie 'username' updated with value: " + a);
    }

    function DeleteCookie() {
      document.cookie = "username=; expires=Thu, 01 Jan 1970 00:00:00 UTC;
path=/";
      alert("Cookie 'username' has been deleted");
    }
  </script>
</body>
</html>
```

**Program 2 : Design a webpage that displays a form that contains an input for user name and password. User is prompted to enter the input user name and password and password become value of the cookies. Write the javascript function for storing the cookies . (04 marks)**

```
<!doctype html>
<html>
  <head>
    <title>Login cookie </title>
  </head>
<body>
  <form>
    <label> Username : </label>
    <input type="text" id="username" required placeholder="enter username">
    <br><br>
    <label>Password : </label>
    <input type="password" id="pass" required placeholder="Enter Password">
    <br><br>
    <input type="button" value="Submit cookie" onclick="StoreCookie()">
  </form>
  <script type="text/javascript">
    function StoreCookie()
    {
      var username=document.getElementById("username").value;
      var password=document.getElementById("pass").value;

      document.cookie="User Name =" +username;
      document.cookie="Password = " +password;

      alert("Cookies have been set for Username and Password!" +username+password);
    }
  </script>
</body>
</html>
```

**Program 3 : Write a webpage that displays a form that contains an input for students rollno and names user is prompted to enter the input student rollno and name and rollno becomes value of the cookie (06marks)**

- (b) Write a webpage that displays a form that contains an input for username & password. User is prompted to entre the input & password & password becomes the value of the cookie. Write a JavaScript function for storing the cookie. It gets executed when the password changes.

```
<!doctype html>
<html>
<head>
    <title>Set Roll Number Cookie</title>
</head>
<body>
    <h2>Enter Student Roll Number and Name</h2>
    <form name="studentForm">
        <label for="rollno">Roll Number:</label>
        <input type="text" id="rollno" placeholder="Enter roll
number"><br><br>

        <label for="name">Student Name:</label>
        <input type="text" id="name" placeholder="Enter student
name"><br><br>

        <button type="button" onclick="storeRollnoCookie()">Set Roll Number
Cookie</button>
    </form>

    <script type="text/javascript">
        function storeRollnoCookie() {
            var rollno = document.getElementById("rollno").value;
            var name = document.getElementById("name").value;

            // Store roll number as a cookie for the current session (no
expiration date)
            document.cookie = "rollno=" + rollno + "; path=/";

            alert("Cookie has been set for Roll Number: " + rollno);
        }
    </script>
</body>
</html>
```