Q1.What is usability testing in web testing?

Ans. Usability testing involves testing of different aspects of any web app like

- It assesses the website’s user friendliness and suitability by gathering.
- The key to usability testing is to study what a user actually does.

Q2.Explain the difference between HTTP and HTTPS?

Ans.Http vs Https:-

	Http	**Https**
Protocol	It is hypertext transfer protocol.	It is a hypertext transfer protocol with secure.
Security	It is less secure as the data can be vulnerable to hackers.	It is designed to prevent hackers from accessing critical information. It is secure against such attacks.
Port	It uses port 80 by default	It uses port 443 by default.
Starts with	HTTP URLs begin with http://	HTTPs URLs begin with https://
Used for	It's a good fit for websites designed for information consumption like blogs.	If the website needs to collect the private information such as credit card number, then it is a more secure protocol.
Scrambling	HTTP does not scramble the data to be transmitted. That's why there is a higher chance that transmitted information is available to hackers.	HTTPS scrambles the data before transmission. At the receiver end, it descrambles to recover the original data. Therefore, the transmitted information is secure which can't be hacked.
Protocol	It operates at TCP/IP level.	HTTPS does not have any separate protocol. It operates using HTTP but uses encrypted TLS/SSL connection.

Domain Name Validation	HTTP websites do not need SSL.	HTTPS requires SSL certificate.
Data encryption	HTTP website doesn't use encryption.	HTTPS websites use data encryption.
Search Ranking	HTTP does not improve search rankings.	HTTPS helps to improve search ranking.
Speed	Fast	Slower than HTTP
Vulnerability	Vulnerable to hackers	It Is highly secure as the data is encrypted before it is seen across a network.

Q3.Write the test scenarios for testing a web site?

Ans. Let’s take an example of ecommerce website

1. Url accessibility
2. Login functionality
3. UI testing
4. Browser compatibility testing
5. Device compatibility testing
6. OS compatibility testing
7. Session creation and ending testing
8. Product description functionality testing
9. Payment testing
10. Cart testing
11. Load testing(fulfilling the criteria of max number of request at a time given by client)
12. Stress testing (To check the upper limit of requests it can handle at a time)

Q4.Write a few Test Cases on GMail functionality.

Ans.https://docs.google.com/spreadsheets/d/1CZY6OSLXPWXOrMP0SX1KRHsU3nExoYV8RpNs3fXdYqs/edit?usp=sharing

Q5.Write any 5 common ATM Machine functionality.

Ans. Functionality of ATM:-

1. Cash Withdrawal
2. Pin generation

3. Balance enquiry
4. Pin Change
5. Slip generation

Q6.Give some examples of web applications that are used in our day to day life.

Ans. 1.Amazon.com

2.Flipkart.com

3.Github.com

4.geeksforgeeks.com

5.twitter.com

Q7.What are the advantages of Using Cookies?

Ans.

1. Cookies are simple to use and implement.
2. Occupies less memory, does not require any server resources and are stored on the user's computer so no extra burden on server.
3. We can configure cookies to expire when the browser session ends (session cookies) or they can exist for a specified length of time on the client’s computer (persistent cookies).
4. Cookies persist a much longer period of time than Session state.

Q8.What is XSS and how can we prevent it?

Ans. Cross-site scripting (also known as XSS) is a web security vulnerability that allows an attacker to compromise the interactions that users have with a vulnerable application. It allows an attacker to circumvent the same origin policy, which is designed to segregate different websites from each other. Cross-site scripting vulnerabilities normally allow an attacker to masquerade as a victim user, to carry out any actions that the user is able to perform, and to access any of the user's data. If the victim user has privileged access within the application, then the attacker might be able to gain full control over all of the application's functionality and data.

Prevention:-

- **Filter input on arrival:-**At the point where user input is received, filter as strictly as possible based on what is expected or valid input.
- **Encode data on output:-**At the point where user-controllable data is output in HTTP responses, encode the output to prevent it from being interpreted as active content. Depending on the output context, this might require applying combinations of HTML, URL, JavaScript, and CSS encoding.
- **Use appropriate response headers:-** To prevent XSS in HTTP responses that aren't intended to contain any HTML or JavaScript, you can use the Content-Type and X-Content-Type-Options headers to ensure that browsers interpret the responses in the way you intend.
- **Content Security Policy:-** As a last line of defense, you can use Content Security Policy (CSP) to reduce the severity of any XSS vulnerabilities that still occur.

Q9.Write a few Cross Browsing Testing TCs for any website.

Ans.Test cases:-

1.Test the login functionality by providing right credentials on chrome.

2. Test the login functionality by providing wrong credentials on chrome.

3.Test the login functionality by providing right credentials on safari.

4.Test the login functionality by providing wrong credentials on safari.

5.Test the login functionality by providing right credentials on firefox.

6.Test the login functionality by providing wrong credentials on firefox.

7.Test the right URL on chrome.

8.Test the wrong URL on chrome.

9.Test the right URL on safari.

10. Test the wrong URL on safari.

11. Test the right URL on firefox.

12. Test the wrong URL on firefox.