How websites work:

A browser makes a request to a server asking for information about a web page. Server responds with data which the browser uses to show the page, or an error code.

There are two major components of a website:

- **Front end(Client-side)**: the way browser renders a website, and clients see it
- **Back end(Server-side)**: a server that processes the request and returns a response

Websites are created using:

- **HTML(Hypertext Markup Language)**: to build the website and define its structure
- **CSS(Cascading Style Sheets)**: adds styling options to change the look of the website
- **JS(JavaScript)**: to implement complex features using interactivity

HTML elements are the building blocks of pages. They define how to display content, for the browser and act like a map.

HTML Structure:

- Doctype definition(<!DOCTYPE html>): Defines page as a HTML5 document. This helps with standardization and tells broswers to use html5 to render the page.
- <html> element: Root element of html page. All elements of the page come after this.
- <head> element: Contains information about the page such as page title.
- <body> element: Defines HTML documents body. Only content inside the body is shown in the browser.
- <h1> heading: Defines a large heading. Increments up to h6 with larger numbers creating smaller headings.
- <p> element: Defines a paragraph or body of text.
- There are various other tags such as buttons, images, lists etc used for different purposes.

Tags:

Tags can contain attributes such as class which can be used to style elements. An elment can have multiple attributes. Elements can also have a unique id attribute which is used for styling and identification by JS.

JS:

JS makes websites interactive. HTML is used to create structure and content, while JS is used to control functionality. No JS would mean a static page and no interactive elements.

JS can dynamically update page in real time. JS can be loaded in the <script> tag or can be remotely loaded by adding the src attribute to the tag.On-click events can be used in elements to execute JS when an event occurs. On-click events can also be defined inside the script tags.

Sensitive data exposure:

Sensitive data exposure occurs when a website does not properly protect or remove sensitive clear text information from the front end source code. Sensitive information can be used to further access different parts of the web app. While testing security for a web app, always review page source for information left there by accident.

HTML injection:

If user input is not sanitized(filtered for malicious input), attackers can inject HTML or JS code into the website/page. As a general rule: never trust user input.To prevent malicious input, a developer could sanitise all user input before passing it to any JS function.