

1. HTML står för HyperText Markup Language, och dess grundläggande syfte är att beskriva och strukturera innehållet på en webbsida genom att använda sig av element och taggar.
2. Ett element är byggstenarna för en webbsida, och är dem som gör så att saker ses på webbsidan. Element exempel: `<p> </p>`. En attribut är egenskaper som används för att ge mer information om ett element. Till exempel, så är `src` i ett `img` element en attribut för `img` elementet.
3. En `<div>` är ett block element medans en `` är ett inline element. En `div` används oftast för att gruppera större delar av innehåll på en webbsida, medans `span` oftast används för att applicera stilmässiga eller funktionella ändringar på en mindre del av en text eller innehållet inuti andra element.
4. Du kan skapa en länk till en annan webbsida men `<a>` elementet. Exempel: `Klicka för att gå till webbsida`
5. En HTML-tabell är som en rutnät där du kan placera information. Det är som ett bord med rader och kolumner. En simpel struktur skapar du med hjälp av följande element: `<table>` (själva border i sig, du placerar resten av elementen i denna), `<tr>` (en rad i border), `<th>` (en rubrikcell) och `<td>` (datacell, där du sätter datan i sig).
6. Med hjälp av `<link>` elementet så kan du linka en extern css fil till en html fil. Du lägger `<link>` elementet i head, och sedan skriver du filepath i href attributen inom link.
7. Inline css är css som du skriver innuti ett html element med hjälp av style attributen. Intern css är när du använder `<style>` elementet innuti en html fil och sedan skriver du css innuti det elementet. Extern css är när du har en separat css fil som du länkar till html filen med hjälp av link, och sedan skriver du all css i den externa css filen.
8. Id och class används i css för att rikta in sig på specifika html element. Id används för specifika html element, för endans ett element får ha ett visst id. Du refererar till html elementets id genom att skriva namnet på id efter ett #, till exempel: `#namnetpåid`. Class fungerar på ett liknande sätt förutom att den kan användas på flera olika element och inte endast på ett enastående. Därför kan en grupp av elements bli påverkade. Du refererar till html elementets class genom att skriva namnet på classen efter en punkt (.), till exempel: `.classnamn`.
9. Du kan göra detta med hjälp av flex. För ger skriver du `display: flex;` på elementet, och sedan kan du lägga till antingen `justify-content: center;` (centrerar horisontellt) eller `align-items: center;` (centrerar vertikalt).
10. Boxmodellen visar hur webbläsaren ritar och beräknar utrymmet för element på en webbsida. Den beskriver också hur varje element ses i form av en rektangulär låda

som omger innehållet och har olika komponenter. De fyra huvudelementen i boxmodellen är: content, padding, border och margin.

11. Bootstrap är en opensource web development framework som finns för att hjälpa utvecklare att snabbt och enkelt bygga responsiv och visuellt fina webbsidor och webbapplicationer.
12. Du använder dig av en link elementet, fast istället för file path till en fil i href, så gör du till bootstraps cdn webbsida, vilket är:
["https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css"](https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css). Efter detta så använder du bootstrap komponenter inom class attributen i elements som vanligt.
13. Bootstrap-griden är ett gridsystem som är uppdelad i 12 kolumner och den gör det enklare att organisera och placera innehåll. Du använder dig av följande namn på classes: container. som skapar en fast breddskontainer, row, som skapar en rad, och till sist col-x (där x står för en siffra). Siffran bestämmer hur mycket plats en column tar upp, från 1 till 12, pga av att gridsystemet är delat i 12 delar. Som exempel, skulle col-6 ta upp halva bredden av gridsystemet pga att 6 är 50% av 12.
14. För att snabbt nämna några grundläggande komponenter som du kan fixa runt med så finns det: buttons, cards, tables, images, navbars och mycket mer, men dessa är endast dem grundläggande.
15. Javascript är ett programmeringsspråk som hjälper html genom att göra sidan med dynamisk och interaktiv. Det finns många skillnader mellan språken, till exempel att html är ett markup language medans Javascript är ett programmeringsspråk. Medans html och css använder sig av klasser och ids, så använder sig javascript av variabler, funktioner och logik. Så helt enkelt används HTML för att strukturera innehåll, CSS för att styra utseendet och JavaScript för att lägga till interaktivitet och dynamik på webbsidor.
16. De förklaras på följande sätt: var: Har function scope, kan deklarerars om och ändras inom samma omfattning. Let: Har block scope, kan inte deklarerars om men kan ändras inom samma omfattning. Const: Har block scope, används för konstanter som inte kan ändras efter tilldelningen.
17. En JavaScript-funktion är ett återanvändbart kodblock som utför en specifik uppgift eller beräkning. Du definerar en funktion genom att skriva "function" och sen namnet på funktionen med paranteser efter som kan innehålla parametrar om man behöver det. Sedan skapar du måsvingar {} för att sedan skriva koden innuti dem. Du anropar en funktion genom att skriva namnet på funktionen men paranteser efter, där argument för funktionen sätts. Till exempel: minFunktion();.
18. Som ett exempel så kommer jag använda onclick attributen till en button som en eventhanterare. Först skapar man funktionen i javascript, men sen måste man skriva

namnet på funktionen (med paranteser) innuti onclick attributen, så att den triggar funktionen när du klickar på knappen.

19. Skillnaden mellan null och undefined är följande: null är ett värde som representerar frånvaron av något värde eller objekt. Du använder den medvetet när du vill representera att ingenting finns. Undefined betyder att en variabel eller objekt har deklarerats men inte har fått ett värde än. Du får också undefined om du inte har ett return värde i en funktion.
20. DOM (Document Object Model) är som en hierarkisk struktur som beskriver och organiserar allt innehåll på en webbsida som om det vore ett träd. I detta träd är varje del av webbsidan (som rubriker, text, bilder, knappar) en "nodboll." Du kan sedan använda detta träd i javascript för att hämta element, ändra innehåll, ta bort element, etc. DOM låter dig alltså kontrollera samt interagera med allt på en webbsida.