

I dessa uppgifter kommer vi att arbeta med webbläsaren snarare än själva html-strukturen. Detta objekt kallas för Browser Object Model (BOM).

Jag rekommenderar att du skapar en ny html/js-fil för dessa övningar.

1. Skapa en knapp i din html. Ge knappen ett id. Syftet med knappen är att öppna en ny flik när användaren klickar på knappen. I js-filen, skapa en window.load-funktion och i denna funktion, skapa en funktion som skall köras när användaren klickar på knappen (addEventListener). I denna funktion skall du öppna ett ny flik, detta görs genom att köra följande funktion:

`window.open(url)`, där url är en sträng med url:n till sidan som skall öppnas. För denna övning skulle rekommenderar jag att du använder samma url som den sida du är på nu.

2. I samma html-fil som i övning 1, skapa en till knapp och ge den ett id. Denna knapps syfte är att stänga den aktuella fliken vid klick. I js-filen, skapa en ny funktion som skall köras vid klick på den nya knappen (fortfarande via addEventListener). I denna klick-funktion vill jag att ni skall köra funktionen:

`window.close()`

3. Webbläsaren innehåller också information kring var någonstans du tidigare har varit. Denna information finns i egenskapen history i objektet window. Skapa en ny knapp i din html med ett id. I js-filen fortsätter vi att lägga till funktioner för knapptryckningar. Denna gång för den nya knappen. Syftet med detta knapptryck är att backa webbläsaren till den senast besökta webbplatsen. I funktionen för knapptryckningen, kör funktionen:

`history.back()`

4. Det finns även en funktion för att gå framåt i historiken. Detta görs med funktionen `history.forward()`. Skapa en ny knapp och se till att klick-funktionen kör funktionen `history.forward()`.
5. Ibland kan det vara av intresse att kunna köra en funktion regelbundet, t.ex. var 10:e sekund eller fördröja en funktion med t.ex. 2 sekunder. Detta görs med funktionerna `setTimeout` och `setInterval`.
I din html-fil, skapa en ny knapp vars syfte är att skapa en tidsinställd funktion.
I js-filen, skapa upp en funktion som körs när du klickar på knappen. I denna funktion, kör funktionen `setTimeout`. Denna funktion har två stycken argument, det första är en funktion som skall köras. Det andra är vilken fördröjning som funktionen skall ha (angett i millisekunder).
Funktionen som du skall köra nu kommer bara att skriva ut `console.log("Funktionen körd")`. Men i andra fall kan du här anropa annan funktionalitet efter behov.
6. För att ändra funktionaliteten från övning fem kan du ändra `setTimeout` till `setInterval` istället. Detta gör att funktionen som anges kommer att köras med det intervall som du anger som andra parameter. Om du testat att uppdatera din sida nu kommer funktionen att köras hela tiden, utan möjlighet att avsluta att funktionen körs om och om igen.
7. Från övning sex ovan. Du behöver lägga till funktionalitet för att kunna avsluta din `setInterval`. För att göra detta kontrollerat, skapa en knapp vars syfte är att stoppa körningen av

funktionen. Skapa en klick-funktion för denna knapp.

I klick-funktionen, kör funktionen `clearTimeout`. Denna funktion kräver ett argument, nämligen resultatet av `setInterval` eller `setTimeout`. Här kommer du behöva klura lite på hur detta kan tänkas göras.

8. Ibland kan det vara bra att känna till vilken webbläsare som används för att besöka din webbplats. Denna information kommer du åt genom `window.navigator`. Testa att skriva ut denna information i konsollen när sidan laddas.
9. En av de roligare sakerna att arbeta med är webbläsarens position i världen. Denna information kommer vi åt från `navigator.geolocation`. Men det är inte alla webbläsare som stöder denna funktionalitet (framför allt äldre webbläsare). Så vi behöver alltid lägga till en kontroll för om webbläsaren stöder `geolocation` innan vi börjar använda denna. När du har gjort denna funktionalitet kan du köra följande funktion:

`navigator.geolocation.getCurrentPosition(showPosition)`, där `showPosition` är en funktion som skall köras när `getCurrentPosition` är klar och har ett resultat.

Skapa funktionen `showPosition` som tar emot ett argument som är ett objekt som innehåller all information kring positionen.

Titta igenom argumentet och skriv ut de egenskaper som du tycker verkar rimliga att använda :)