Laboration 3 - Ritprogram

Er uppgift är att skriva en komplett webbsida med HTML, CSS och JavaScript. Laborationen ska utföras av en eller två personer. Utveckla gärna i JS Bin eller Brackets/Sublime/NotePad++.

Laborationen är obligatorisk, men vissa delar är markerade med "VG". Om man är nära en gräns på tentan kan gjorda VG-labbar höja kursbetyget.

# Inlämning

Koden ska läggas upp på er Github-sida.

Laborationen ska lämnas in **senast 2016-12-01** genom att skicka ett meddelande till David via Slack med länken till er sida. Försenad inlämning kan leda till försenat kursbetyg.

# Uppgift

Uppgiften handlar om att använda DOM-manipulation, Events och prototyper, samt rita på Canvas. Om ni har gjort Lab 2 rätt så ska ni kunna kopiera koden utan att göra ändringar.

Ni ska göra en webbsida som ska innehålla följande:

* ett element som man kan klicka på för att visa en meny
* en canvas som man ska kunna rita på
* ett select-element för att välja färg på linjerna man ritar
* ett textfält för att skriva in nya färger
* en "status bar", ett valfritt element som visar en kort beskrivning av vad saker på webbsidan gör, eller vad man håller på med
* knappar för att välja om man vill rita en cirkel, rektangel eller triangel
* VG: knapp för polygon
* en knapp för att avbryta ritning av någon av figurerna
* en knapp för att rensa canvas-elementet
* en knapp för att exportera det man ritat till JSON
* VG: en knapp för att importera JSON

När det gäller CSS har ni konstnärlig frihet att göra det hur snyggt som helst.

## Meny

Det ska finnas ett element som motsvarar en meny. Det ska bete sig precis som menyer gör i vanliga program. När man klickar på det ska det dyka upp en lista där man kan välja: rensa canvas, rita cirkel/rektangel/triangel (polygon), exportera till JSON.

## Status bar

Ett element ska bete sig som en "status bar" i vanliga applikationer. När muspekaren befinner sig över ett element som man kan klicka på, ska det stå vad som kommer hända om man klickar på elementet. Exempel: för knappen som börjar rita en triangel kan det stå "Starts drawing a triangle".

När muspekaren befinner sig över canvas-elementet ska det stå i status bar vad som är nästa steg i utritningen.

## Canvas

### Triangel

För att rita en triangel behöver man tre punkter. När användaren klickar på knappen för att rita triangel ska programmet förvänta sig att användaren klickar tre gånger i canvas-elementet. Varje gång användaren klickar ska positionen sparas. När tredje klicket äger rum ska hela triangeln ritas ut. Rita gärna ut delar av triangeln tidigare, så användaren ser var hen har klickat. Under hela processen ska det framgå i status bar att användaren ska klicka i canvas-elementet. *Använd Triangel-objektet som ni gjorde i Lab 2!*

### Rektangel

För att rita en rektangel behöver man bara två punkter, annars ska det bete sig på samma sätt som för en triangel.

### Cirkel

För att rita en cirkel behöver man en mittpunkt och en radie. Precis som för en rektangel behöver användaren klicka två gånger. Den första gången är mittpunkten och den andra är en godtycklig punkt på radien. (Ni blir alltså tvungna att räkna ut radien genom att ta avståndet mellan punkterna.)

### VG: Polygon

Eftersom en polygon kan ha olika många punkter så löser vi det genom att knappen för att börja rita polygon byter text och funktion när användaren klickat på den, från "Rita polygon" till "Avsluta polygon". Efter att användaren klickar på knappen första gången kan man klicka hur många gånger man vill i canvas. Först när användaren klickar på knappen "Avsluta polygon" skapas och ritas polygonen ut. Sedan ska knappen "Avsluta polygon" återgå till "Rita polygon".

## Välja färg

Select-elementet ska ha ett litet antal fördefinierade färger. Det första elementet ska ha texten "Välj färg" och inte vara valbart. Den färg som är vald i select-elementet ska användas när en ny figur ska ritas.

Det ska finnas ett textfält där användaren kan skriva in en HTML-färg i hexadecimalt format. Exempel: "#c080ff". Man ska kunna klicka på en knapp för att lägga till färgen i select-elementet. Om värdet i textfältet inte är en giltig HTML-färg ska användaren informeras på lämpligt sätt och knappen för att lägga till färgen ska vara disabled.

(Namn på färger som "hotpink", "beige" osv. behöver ni inte stödja.)

## JSON

När användaren klickar på knappen eller menyalternativet "exportera till JSON" ska de objekt som ritats ut omvandlas till JSON och läggas i en textruta någonstans på sidan. Ni behöver lägga till en egenskap för linjefärg till era figurer.

VG: Lägg till en knapp för att importera JSON. Användaren ska klistra in JSON-kod som ni gör om till en lista med figurer när användaren klickar på knappen. Sedan ska figurerna ritas ut, i ordning.

## VG: Ändra figurer med formulärelement

När användaren klickar i canvas ska ni kontrollera om det ligger en figur på (tillräckligt nära) den positionen. Om det finns en figur där ska ni skapa formulärelement: två stycken textfält om det är en cirkel eller en rektangel, tre om det är en triangel, så många som behövs för en polygon. Nu ska användaren kunna skriva in nya värden i textfälten. Figuren ska uppdateras och ritas om med de nya värdena.

Glöm inte att status bar ska visa information för alla element.

Kom ihåg att lägga upp webbsidan på er Github-sida!

Om ni är två så går det bra att ha en gemensam sida.