# Plantegning

## 1.0 – introduktion

Plantegningen er et SVG billede som automatisk generes på hjemmesiden når sælger åbner en bestilling. Tegningen indeholder alle komponenter, på nær tagbelægning, midterspær, væg belægninger og lister. En plantegning ser ud som eksemplet nedenform, men varierer alt efter carportens mål og tagrejsningsvalg.

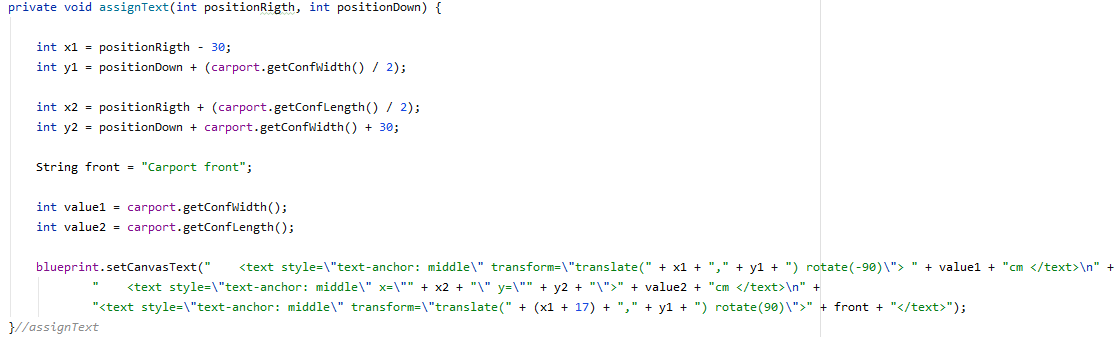
## 1.1 – Blueprint opbygning

Blueprint klassen indeholder en række attributter som hver indeholder et lille tekst udsnit af det samlede SVG billede. Nogle attributter er strings, mens andre er lister af strings (ArrayList <String>).  
En særlig attribut er blueprintSVG, den indenholder den færdige string som resten af attributterne puttes ind i.

## 1.2 – SVG elementer

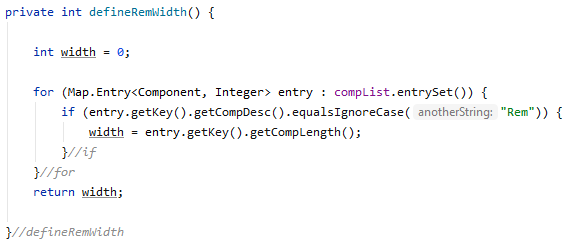
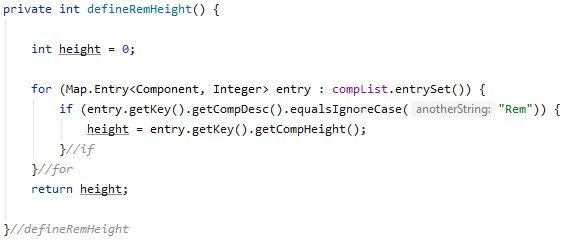
Hvert attribut er tom fra starten, men fyldes op af metoder når et offerRequest objekt bliver skabt. De skitserelaterede elementer, så som kanvas, pile, tekst og lignende bliver skab af en enkelt(dog individuel) metode, hvor carportens dele, så som stolper og remme, skal igennem flere metoder som definerer komponentens højde, bredde og placering.

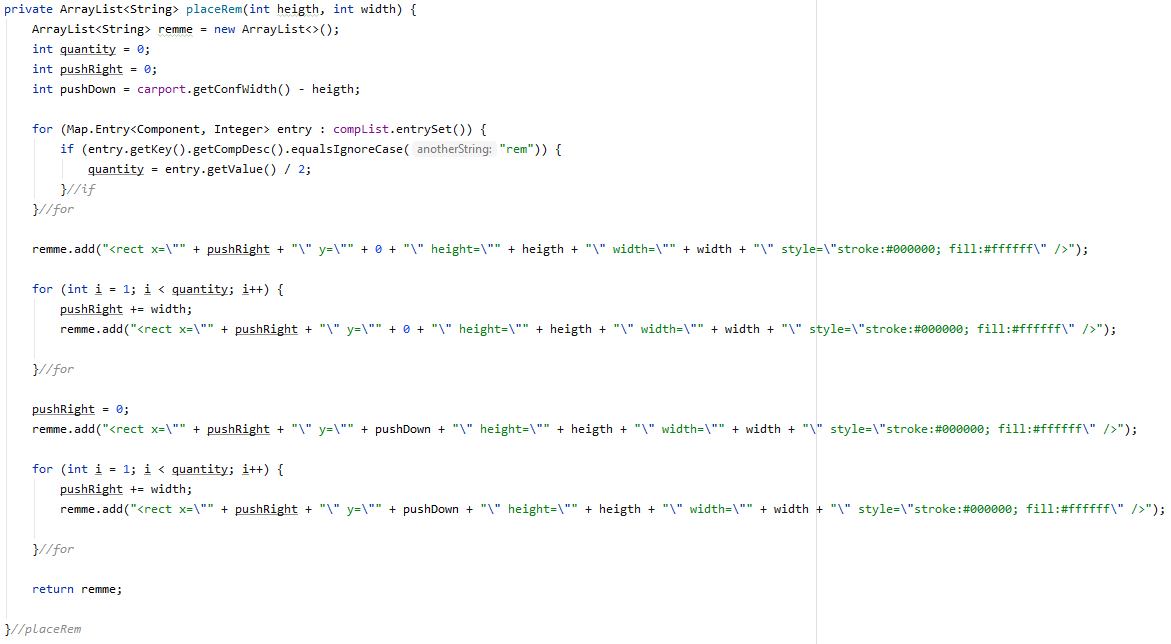
## 1.2.1 – simpelt element

Et eksempel på et simpelt element kan være den tekst som indikerer carportens front.  
Vi betragter koden nedenfor. Først har vi skrevet hele SVG teksten og sat den ind i en setter metode for den tilhørende attribut i Blueprint klassen.   
Herefter har vi identificeret de dele af SVG koden som skal være dynamisk og udskiftet værdier med variabler.   
Disse variabler sætter vi så til at tage imod data fra carport objektet eller som statiske data, hvilke skal være overskuelige at rette ved behov.

## 1.2.2 – carport element

Carport elementerne fungerer på samme måde som de simple elementer, bortset fra at vi er nød til at definerer målene på den figur som repræsenterer komponenterne.  
Vi begynder med en lille start-metode som opsamler dimensionerne og bruger dem i den endelige metode.

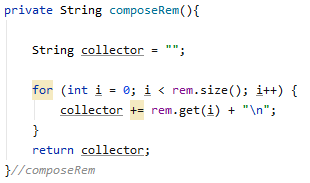
Illustrationer af de to define-metoder:

Illustration af den endelige metode, som sender en liste med strings tilbage til den lille startmetod, som sender den videre med en setter metod.  
Hvad er anderledes her i forhold til et simpelt element, er dette består af et antal af det samme element som varierer og skal stå forskellige steder i kanvasset.

## 1.3 komponering af SVG string



BlueprintSVG attributten er som de andre, tom fra starten, men har en metode som arrangerer og implementerer de nu fyldte tekst elementer.

De attributer som er lister, har dog behov for at blive pakket ud, så de har deres egne compose-metoder.

## 1.4 sammenkobling

For at få hele møllen til at køre, har vi en metode som kronologisk kalder de individuelle elementers metoder og til sidst sætter den færdige tekst værdi ind i blueprintSVG variablen.

Denne metode sætter også de overordnede mål for billede kanvasset og forholdet til carport kanvasset.

## 1.5 resultat

Her vises indholdet af blueprintSVG string værdien når programmet er kørt og har genereret alle målene og placeringerne på elementerne:

(resultat del 1/2)



(resultat del 2/2)

