# ETS tabellen

Binary Tree -> gesorteerde ETS tabel op basis van veld dat de sleutel is.

Hash map-> ETS tabel

* Keuze welke van de twee hangt af van uw tabel

{{key, make.ref()},……………..}

{…,…,…} < {…,…,…} -> tullips vergelijken met elkaar  
Key kan tijd zijn voor bv een agenda en is ook de meest beduidende waarde.

# Patern matching

<<…, V = 20,...>> “GUARD” 🡪 code uitvoeren   
Door deze patern matching ga je geen if og case structuur nodig hebben  
V -> f(…,V)

F(…,V) ->;  
f(…,W)->;  
f(…,Z)->.

Vaak schrijf je ergens een variabele: f(if V then….) -> dit is pattern matching Moest dit niet voorkomen, er is geen V in f, dan crashed het process oor processen die je zelf schrijft moet je spawn link gebruiken, als er dan een process crashed, crashed het programma Anders zal het programma doorlopen en besef je misschien niet dat die gecrashed is.

# Toepassing uitwerken, event manager

We hebben een agenda waarin je de tijdstip in zet {{tijdstip,make.ref()},…,…,…,…} Van het moment dat de tijd het tijdstip inhaalt moet je een bericht weergeven in het logboek.

Tip:

Receive  
 {post,…} -> in kalender steken

After ?

End

Eerste stap is een kalander waar we in kunnen opslaan, event manager maken die dan nieuwe events kan toevoegen en dan een after toont die hierboven is uitgelegd.

Begrens(G) ->  
 H = if  
 g>100->100;  
 true ->G,

H.