

## Beskrivning

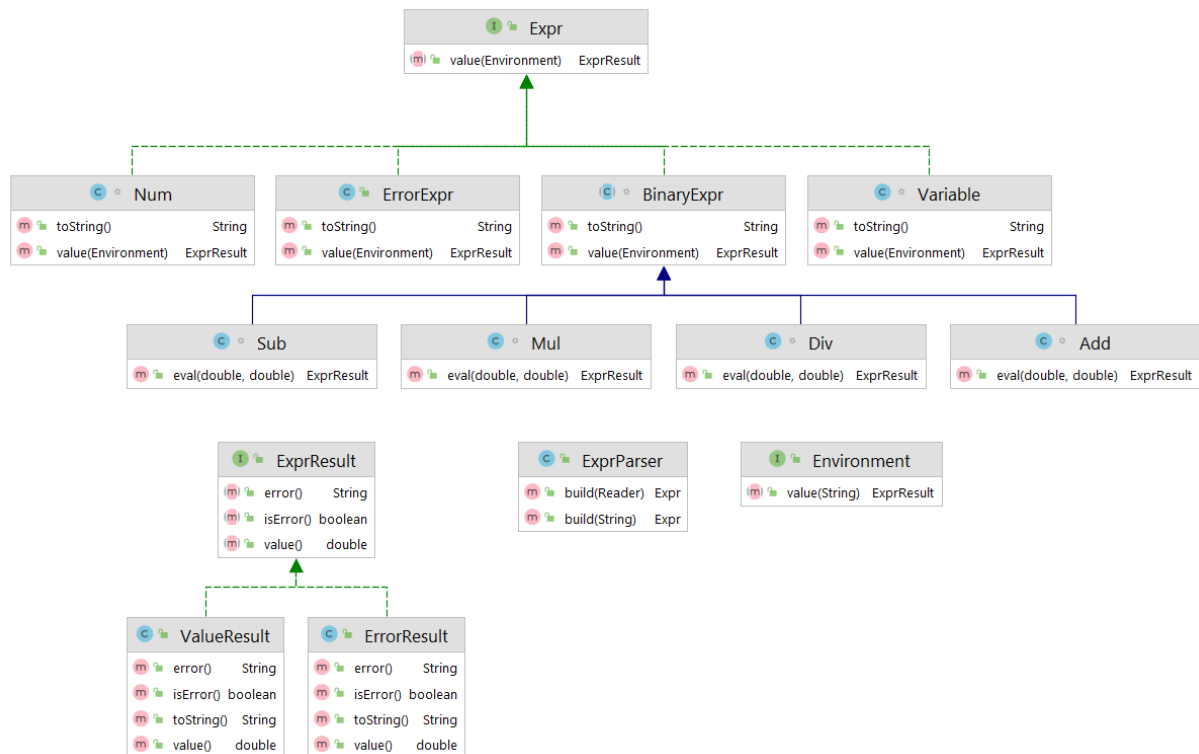
De designmönster vi använt är observer, factory och strategy. Observermönstret används i klassen med namnet Observer. Observatören, i detta fallet XL registreras inuti XLModel-objektet som är det objekt som lyssnas på. Vid en tillståndsuppdatering meddelas observatören via notifyObservers-metoden. Fabriksmönstret används i klassen CellFactory. CellFactory har en enda enkel metod för att utifrån en sträng avgöra vilken sorts cell som ska tillverkas och därefter tillverka den. Slutligen används strategy då funktionaliteten varierar, i detta fall vid olika sorters input. Beroende på input skapas olika slags celler.

När det kommer till goda designprinciper så har vi givetvis tagit hänsyn till de allra flesta av de principer som kursen lär ut. Vi har undvikit att upprepa kod och vill påstå att vår kod därmed följer don't repeat yourself principle. Att alla celler implementerar interfacet Cell möjliggör tillägg av fler celler i framtiden. Vi har således tagit open closed principle i beaktande när vi utvecklat koden. The single responsibility principle är ytterligare en princip vi strävat efter att följa.

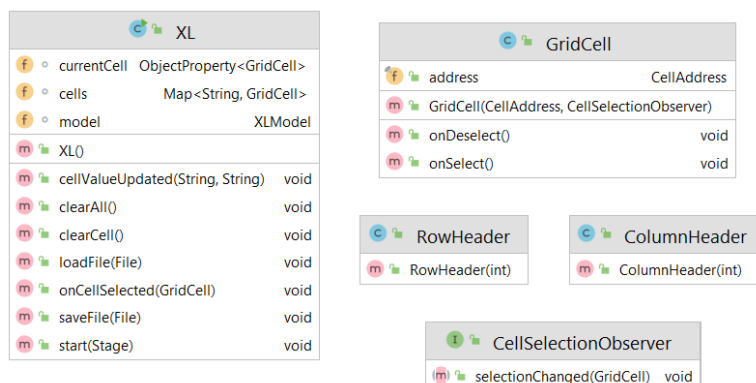
# Klassdiagram

Klasserna syns här uppdelade i sina respektive paket.

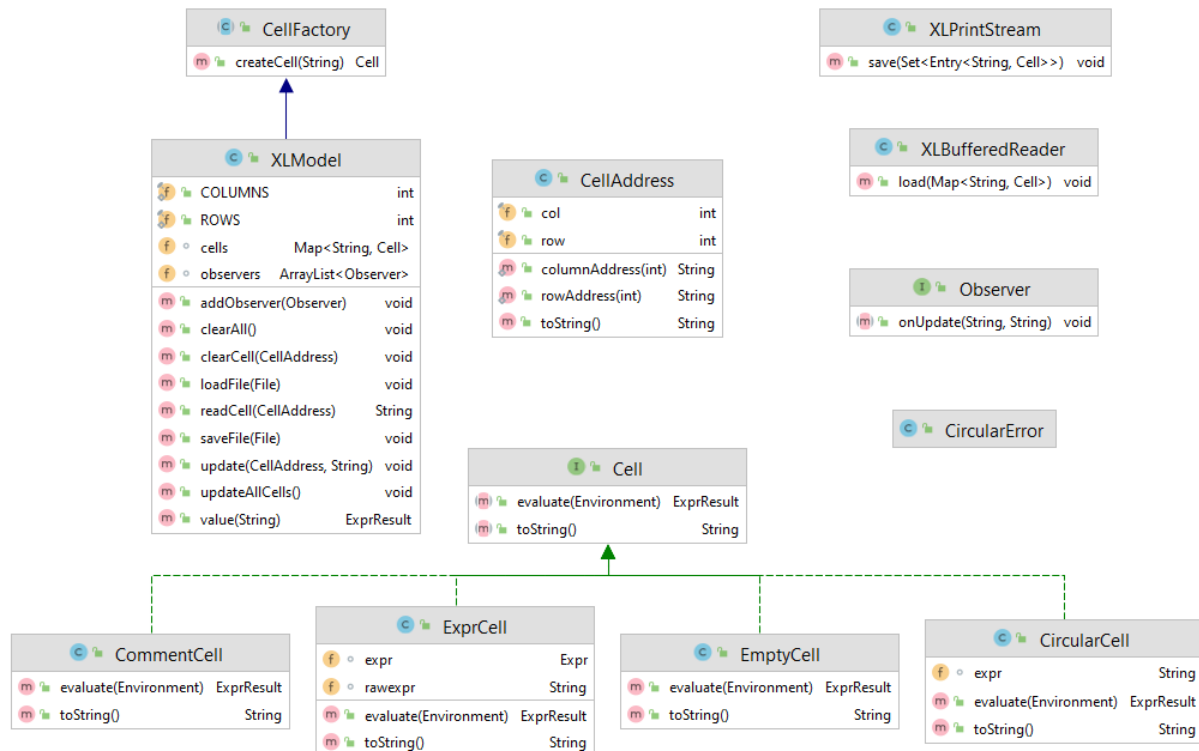
expr:



gui:



model:



util:

