

Beskrivning XL - Grupp 8

MVC

För att utveckla XL har vi följt mallen för MVC. Vi har låtit modellen utföra alla ändringar, och hållt vyn oerhört simpel för att endast *visa resultatet från modellen*.

Nya klasser

Vi har lagt till följande klasser:

- **ModelObserver** - Ett gränssnitt som man kan använda för att observera ändringar i modellen. **XL**-klassen använder denna med hjälp av lambdas, vilket gör det lätt och smidigt att lägga till & skapa observatörer.
- **ObservableModel** - Ett gränssnitt som **XLModel** implementerar för att notifiera observatörer. Dessa gränssnitt använder *observatörsmodellen*

-
- **XLModel** - Denna klass representerar ett kalkylark och innehåller all data.

Representationen av en cell.

Vi valde att representera en cell som en sträng, dvs vi sparar det *råa* sträng-värdet, och beräknar dess resultat, varje gång den behövs i vyn. Motiveringen till detta är att detta blev resultatet av vår första iteration och ansågs enkelt och intuitivt. Datastrukturen för att hålla rätt på celler i **XLModel** är en mapp enligt: **Map<String, String>**, där strängarna är: *Address -> Raw string value*. En nackdel med att beräkna resultatet varje gång vyn behöver det är att det kan ta väldigt lång tid vid rekursion och större ark. Eftersom det inte ställdes några prestandakrav så ansågs detta inte vara något problem, men det bör påpekas; att om det vore *på riktigt*, borde en bättre algoritm användas, exempelvis genom att cacha värdet med en hjälpklass.