

# Ticket sales

Programmet skapades i samband med instruktionerna vi fick gällande uppgiften. Målet var att skapa ett system för ett företag som hanterar försäljning av konsert biljetter.

## Beskrivning:

Till en början skapade jag bara en simpel **while (true)** loop som skulle fungera som en meny som fortsätter köras och samlar ihop alla olika uppgifter för lätt åtkomst. Efter det så började jag tänka på vad för slags kommandon jag ville ha samt vad som krävdes för uppgiften. När jag hade bestämt mig så började jag implementera de en efter en.

Hjälp och Exit) Hjälp var det första jag implementera och jag skrev en uppmuntrande text för att hjälpa användaren hitta kommandolistan. Exit lades till direkt efter.

Köp biljett) Jag ville göra det så klart som möjligt för användaren att faktiskt köpa biljetter så jag började med att radda upp alla konserter och sen be användaren ange vald konsert. Jag såg till att det spelade ingen roll om man använde versaler eller gemener samt att om en ogiltig konsert angavs så ges ett meddelande. Sen bad jag om namn och ålder, fast istället för att bara lagra åldern direkt i en string så gjorde jag en **try**. Det såg till att endast **ints** ledde en vidare, sen beroende på ålder så bestämde jag fram kostnaden och sen lades kunden in i en lista som hade alla inkludera alla kundens attribut. Efter det sparade jag all info men det går jag in på mer senare.

Biljett Lista) Det är en simpel lista som alla kunder sätts in i och den går att få ut printad med samtliga attribut på kunden

Konsert stats) Den kollar igenom ovan nämnda listan och listar ut hur många biljetter har sålts av varje konsert. den gör det genom en foreach loop som höjer summan av de globala konsert variablarna med en för varje såld biljett.

Vinster) Kollar igenom alla sålda biljetter och checkar vilken konsert den tillhör till, åldersgruppen därmed kostnaden samt antal sålda biljetter sålda både totalt samt individuellt för konserterna.

Återköp och Återköp historia) Det ber användaren först att namnge vilken konsert biljetten de vill återbetala är till sen order nummret. Därefter antecknas biljettens åldersgrupp och därmed kostnaden och uppdaterar de globala variablarna med

resultaten. Genom **återköp historia** kan man sedan checka antalet återköp både totalt och specifika åldersgrupper samt summan pengar som har återbetalats. Ladda och Spara) Det gör så att alla attribut från kund klassen av kunder som köper biljetter sparas på ett text dokument som kan laddas upp vid start(jag gjorde en meny för det för att ge användaren valet att ladda upp gammal spar data eller för att köra som om det vore nytt).

## Hur det gick:

Jag skulle säga att det gick mycket bättre med denna uppgiften över myrstacken. När jag gjorde myrstacken så fattade jag inte allt helt klart än men med denna uppgift kände jag mig mycket säkrare och mer självsäker. Jag utgick en del från myrstacken när jag skrev programmet fast gjorde sakerna på en lite mer avancerad nivå samt mer effektivt, så känns det iaf. Jag fastnade rätt ofta med myrstacken, jag fastnade en del med denna också fast inte nära lika mycket. Det främsta jag fastnade på var hela spara och ladda grejen så jag fick ta och fråga Jakob om hjälp med det men det känns som jag uppfattar det hyfsat bra nu. Ett annat problem jag hade var att få order nummern att inte återställas varje gång programmet stängs ned och öppnas. Tyvärr listade jag inte ut hur man skulle kunna lösa det så programmet blir mycket mindre effektivt efter första användningen. Så länge man inte stänger ned det så borde allt funka perfekt dock.

Jag tror hela programmet skulle kunna vart mycket bättre ifall jag inte gjorde det så sent och stressade fram det plus ifall jag försökte förstå mer grejer på lektionerna istället för att kolla på youtube tutorials. Dock så lärde jag mig en del från dina lektions koder bara att det tog ett litet tag att förstå allt. Den största förbättringen jag skulle vilja göra ifall jag hade tid och kunskapen är att göra hela grejen mer effektiv. tex. tillåta flera biljetter köpas på samma gång, göra återköps systemet mer detaljerat eftersom nu är det rätt lätt att "förstöra" det pga tidigare nämnda problem med hela spar/ladd grejen.