

Specifikation

Inledning

Programmet ska utifrån en lista med information kring olika teatrar uppdatera varje kväll hur många biljetter som har sålts till föreställningen och hur stor procent av platserna som blev sålda. Därefter ska teatrarna sortera utifrån hur bra de har sålt. Användaren ska då få skriva in antal sålda biljetter i varje prisklass för varje teater. Uppgiften utmaningar och svårigheter blir att skapa ett användargränssnitt eftersom jag aldrig använt användargränssnitt tidigare och att hålla koll så all information hamnar hos rätt teater och ifall informationen är rimlig då det kan lätt bli fel.

Användarscenarier

Användaren 1 kommer till en meny och väljer "statistik" där de får upp tre olika teatrar och ska kunna klicka på de olika teatrarna och därefter fylla i hur många biljetter som har sålts för varje prisklass. Exempelvis 20 barn biljetter, 50 vuxenbiljetter och 5 pensionärsbiljetter för teater 1. Därefter kan användare 1 välja att fortsättas fylla i de andra teatrarna eller se resultatet för dagen. Användare 1 väljer då att se resultatet för dagen och får upp en tabell med de olika teatrarna där de är rankade efter hur mycket biljetter de sålt i förhållande till platserna. Därefter kan användare välja att fortsätta mata in biljettförsäljningen för flera dagar eller avsluta

Användare 2 väljer att klicka på knappen "sålda biljetter" och får sedan välja en teater att se statistik över hur många biljetter som har sålts av varje sort.

Programskelett:

```
class Teater:
    """Beskriver en teater, har attributen namn, antal platser, vuxenpris, pensionärspris, barnpris, intäkt, sålda platser och belägg"""

    def __init__(self, namn, platser, vuxen, pensionär, barn, prisintäkt = 0, sålda_platser = 0, belägg = 0):
        """Konstruktorn som anropas när vi skapar en ny teater
        Parametrar: namn, antal platser, vuxenpris, pensionärspris, barnpris, intäkt, sålda platser, belägg
        """

    def ändra_intäkt(self, platser, prisgrupp):
        """adderar intäkten i attributet prisintäcker
        Parametrar: self, platser, prisgrupp
        Returnerar: inget """

    def ändrar_sålda_platser(self, platser):
        """ändrar attributet antal sålda platser
        Parametrar: self, platser
        Returnerar: inget"""

    def beräkna_belägg(self):
        """Beräknar belägg och ändrar attributet belägg
        Parametrar: self
        Returnerar: inget"""

def hitta_siffror(rad):
    """Hittar siffror från raden
    Parametrar: rad
    Returnerar: siffrorna (int)"""
```

```

class Föreställning:
    """Beskriver en föreställning"""

    def __init__(self):
        """ Skapar en tom lista där teatrarna ska läggas in """

    def las_teatrar(self):
        """ Läser teatrar från filen till attributet teatrar.
        Parametrar: self
        Returnerar: inget """

    def statistik(self):
        """ Skriver ut statistiken om alla teatrar per föreställning
        Parametrar: self
        Returnerar: inget """

    def visa_alla(self):
        """ Skriver ut information om alla teatrars bijettpreiser per föreställning
        Parametrar: self
        Returnerar: inget """

    def hitta_namn(self):
        """ Hanterar insättning
        Parametrar: self
        Returnerar: Inget"""

def menyval():
    """Skriver ut menyn, läser in och returnerar användarens val
    Parametrar: inget
    Returnerar: val"""

def main():
    """Huvudprogrammet, som skapar föreställning-objektet, läser in från fil
    anropar menyn och hanterar inmatningen som gjorts.
    Parametrar: inga
    Returnerar: inget"""

if __name__ == "__main__":
    main()

```

Minnet:

Det kommer pågå oändligt med föreställningar beroende på användarens val. Men för varje föreställning så kommer minnet att se ut såhär:

