class Sang:

        def \_\_init\_\_(self,artist,tittel):

            self.\_artist = artist

            self.\_tittel = tittel

        def spill(self):            # Metoden spill() skriver ut meldingen i terminalen

            print("Spiller " + self.\_artist + " => " + self.\_tittel)

        def sjekk\_artist(self, navn):   # parameter 'navn'

            first\_list = []         # skal bestå av navn

            second\_list = []        # skal bestå av list av self.\_artist

            for i in navn.split():

                first\_list.append(i)

            for i in self.\_artist.split():

                second\_list.append(i)

            count = 0

            while count < len(first\_list):

                if first\_list[count] in second\_list:    # dersom en item i liste er også inne i liste som består av self.\_artist

                    return True

                else:

                    count += 1

            return False      # hvis ingen elementer samsvarer

        def sjekk\_tittel(self, tittel):

            sang\_tittel = self.\_tittel.lower()

            sang\_tittel2 = tittel.lower()

            if sang\_tittel == sang\_tittel2:     # dersom tittel er lik med tittel fra Sang objekt

                return True

            else:

                return False

# sjekker både artist og tittel av sang ved å kalle metoder sjekk\_artist og sjekk\_tittel

        def sjekk\_artist\_og\_tittel(self, navn, tittel):

            if self.sjekk\_artist(navn) == True and self.sjekk\_tittel(tittel)== True:

                return True

            else:

                return False

class Spilleliste:

    def \_\_init\_\_(self, listenavn):

        self.\_sanger = []

        self.\_navn = listenavn

# leser inn data fra "filnavn", og oppretter Sang-objekter. Deretter appender disse objektene til " \_sanger "

    def les\_fil(self, filnavn):

        for line in open(filnavn, "r"):

            kolon = line.strip().split(";")

            alle\_data = Sang(kolon[1], kolon[0])

            self.\_sanger.append(alle\_data)

# metoden legger ny\_sang til "\_sanger"

    def legg\_til\_sang(self, ny\_sang):

        self.\_sanger.append(ny\_sang)

# Metoden fjerner sang fra "\_sanger"

    def fjern\_sang(self,sang):

        self.\_sanger.remove(sang)

# Metoden spiller sanger ved å kalle metoden spill() fra klassen "Sang"

    def spill\_sang(self, sang):

        sang.spill()

# Metoden spiller alle sanger (inkludert ny\_sang) fra self.\_sanger ved å kalle metoden spill()

    def spill\_alle(self):

        for item in self.\_sanger:

            item.spill()

    def finn\_sang(self,tittel):

        for sang in self.\_sanger:

            if sang.sjekk\_tittel(tittel) :      # sjekker om tittel er like ved å kalle metoden sjekk\_tittel

                return sang         # dersom tittel er lik, returnerer sang

        return None                 # dersom ingen tittel er like

    def hent\_artist\_utvalg(self, artistnavn):

        utvalg\_sanger = []

        for songItem in self.\_sanger:   # kaller alle sang objekter fra self\_sanger

            if songItem.sjekk\_artist(artistnavn) == True:  # kaller sjekk\_artist for å sjekke

                utvalg\_sanger.append(songItem)

        return utvalg\_sanger

print(Spilleliste)