

```

class ListePersonnes:
    def __init__(self):
        self.liste_personnes = []

    def ajouter_personne(self, nom, age):
        if isinstance(age, int) and age > 0: # Vérification de l'âge
            valide
            self.liste_personnes.append(self.Personne(nom, age))
            print("Personne ajoutée avec succès.")
        else:
            print("L'âge doit être un entier positif.")

    def afficher_personnes(self):
        if self.liste_personnes:
            for personne in self.liste_personnes:
                print(personne)
        else:
            print("La liste est vide.")

    def rechercher_personne(self, nom):
        personnes_trouvees = [personne for personne in
self.liste_personnes if personne.nom == nom]
        if personnes_trouvees:
            print("Personnes trouvées :")
            for personne in personnes_trouvees:
                print(personne)
        else:
            print(f"Aucune personne nommée '{nom}' trouvée dans la
liste.")

    def filtrer_personnes_par_age(self, min_age, max_age):
        personnes_filtrees = [personne for personne in
self.liste_personnes if min_age <= personne.age <= max_age]
        if personnes_filtrees:
            print("Personnes dont l'âge est compris entre", min_age,
"et", max_age, ":")
            for personne in personnes_filtrees:
                print(personne)
        else:
            print(f"Aucune personne dont l'âge est compris entre
(min_age) et (max_age) n'a été trouvée.")

```

```

class FileAttente:
    def __init__(self):
        self.file_attente = []
        self.file_prioritaire = []

    def ajouter_personne_en_attente(self, nom):
        self.file_attente.append(nom)
        print(f"{nom} a été ajouté à la file d'attente.")

    def ajouter_personne_prioritaire(self, nom):
        self.file_prioritaire.append(nom)
        print(f"{nom} a été ajouté en tant que personne prioritaire.")

    def supprimer_personne_de_attente(self):
        if self.file_prioritaire:
            personne_supprimee = self.file_prioritaire.pop(0)
            print(f"{personne_supprimee} a été supprimé de la file
d'attente prioritaire.")
        elif self.file_attente:
            personne_supprimee = self.file_attente.pop(0)
            print(f"{personne_supprimee} a été supprimé de la file
d'attente.")
        else:
            print("La file d'attente est vide.")

```

```

import recherche_list as rl
from devoirpoo import FileAttente as F

def main():
    liste = rl.ListePersonnes()
    file_attente = F.FileAttente()

    while True:
        print("\nMenu principal :")
        print("1. Ajouter une personne")
        print("2. Afficher les personnes")
        print("3. Rechercher une personne par nom")
        print("4. Filtrer les personnes par âge")
        print("5. Gérer la file d'attente")
        print("6. Quitter")

        choix = input("Entrez votre choix : ")

```

```

        if choix == "1":
            nom = input("Entrez le nom de la personne : ")
            while True:
                age_input = input("Entrez l'âge de la personne : ")
                try:
                    age = int(age_input)
                    if age < 0:
                        print("Veuillez entrer un âge positif SVP !")
                    else:
                        liste.ajouter_personne(nom, age)
                        break
                except ValueError:
                    print("Veuillez entrer un âge valide (entier).")
            elif choix == "2":
                print("\nPersonnes dans la liste :")
                liste.afficher_personnes()
            elif choix == "3":
                nom = input("Entrez le nom de la personne à rechercher : ")
                liste.rechercher_personne(nom)
            elif choix == "4":
                min_age = int(input("Entrez l'âge minimum : "))
                max_age = int(input("Entrez l'âge maximum : "))
                liste.filtrer_personnes_par_age(min_age, max_age)
            elif choix == "5":
                F.gestion_file_attente(file_attente)
            elif choix == "6":
                print("Programme terminé.")
                break
            else:
                print("Choix invalide.")

if __name__ == "__main__":
    main()

```

```

class SalleCinema:
    def __init__(self, capacite, places_speciales):
        self.capacite = capacite
        self.places_disponibles = capacite
        self.places_reserves = {}
        self.places_speciales = places_speciales

```

```

        self.places_speciales_reserves = {}

    def reserver_place(self, nom, place):
        if place in self.places_reserves or place in
self.places_speciales_reserves:
            print(f"La place {place} est déjà réservée.")
        elif place <= self.capacite:
            if nom in self.places_reserves.values() or nom in
self.places_speciales_reserves.values():
                print(f"{nom} a déjà réservé une place.")
            else:
                if place <= self.capacite - len(self.places_reserves) -
len(self.places_speciales_reserves):
                    if place > self.capacite -
len(self.places_reserves):
                        self.places_speciales_reserves[place] = nom
                    else:
                        self.places_reserves[place] = nom
                        self.places_disponibles -= 1
                        print(f"{nom} a réservé la place {place}.")
                else:
                    print("Désolé, plus de places disponibles.")
        else:
            print("Numéro de place invalide.")

    def afficher_places_reserves(self):
        print("Places réservées :")
        for place, nom in self.places_reserves.items():
            print(f"Place {place}: {nom}")
        print("Places spéciales réservées :")
        for place, nom in self.places_speciales_reserves.items():
            print(f"Place spéciale {place}: {nom}")

    def nombre_places_disponibles(self):
        print(f"Il reste {self.places_disponibles} places
disponibles.")

    def filtrer_reservations_par_personne(self, nom):
        reservations = {place: nom for place, n in
self.places_reserves.items() if n == nom}
        reservations.update({place: nom for place, n in
self.places_speciales_reserves.items() if n == nom})
        if reservations:

```

```

        print(f"Réservations pour {nom} :")
        for place, nom in reservations.items():
            print(f"Place {place}")
    else:
        print(f"Aucune réservation pour {nom}.")

    def annuler_reservation(self, nom):
        removed = []
        for place, n in list(self.places_reserves.items()):
            if n == nom:
                removed.append(place)
                del self.places_reserves[place]
                self.places_disponibles += 1
        for place, n in list(self.places_speciales_reserves.items()):
            if n == nom:
                removed.append(place)
                del self.places_speciales_reserves[place]
                self.places_disponibles += 1
        if removed:
            print(f"Les réservations pour {nom} ont été annulées pour les places suivantes : {removed}.")
        else:
            print(f"Aucune réservation trouvée pour {nom}.")

    def reserver_place_speciale(self, nom):
        for place in range(1, self.places_speciales + 1):
            if place not in self.places_reserves and place not in self.places_speciales_reserves:
                self.places_speciales_reserves[place] = nom
                self.places_disponibles -= 1
                print(f"{nom} a réservé une place spéciale (place {place}).")
                return
        print("Désolé, plus de places spéciales disponibles.")

```

```

from reservation import SalleCinema

def afficher_menu():
    print("\nMenu principal :")
    print("1. Réserver une place")
    print("2. Afficher les places réservées")

```

```

    print("3. Afficher le nombre de places disponibles")
    print("4. Filtrer les réservations par personne")
    print("5. Annuler une réservation")
    print("6. Réserver une place spéciale")
    print("7. Quitter")

def main():
    salle = SalleCinema(capacite=50, places_speciales=5)

    while True:
        afficher_menu()
        choix = input("Entrez votre choix : ")

        if choix == "1":
            nom = input("Entrez votre nom : ")
            place = int(input("Entrez le numéro de la place : "))
            salle.reserver_place(nom, place)
        elif choix == "2":
            salle.afficher_places_reserves()
        elif choix == "3":
            salle.nombre_places_disponibles()
        elif choix == "4":
            nom = input("Entrez le nom de la personne : ")
            salle.filtrer_reservations_par_personne(nom)
        elif choix == "5":
            nom = input("Entrez le nom de la personne dont vous
souhaitez annuler la réservation : ")
            salle.annuler_reservation(nom)
        elif choix == "6":
            nom = input("Entrez votre nom : ")
            salle.reserver_place_speciale(nom)
        elif choix == "7":
            print("Programme terminé.")
            break
        else:
            print("Choix invalide. Veuillez entrer un nombre entre 1 et
7.")

if __name__ == "__main__":
    main()

```

```

class FileAttente:
    def __init__(self):
        self.file_attente = []
        self.file_prioritaire = []

    def ajouter_personne_en_attente(self, nom):
        self.file_attente.append(nom)
        print(f"{nom} a été ajouté à la file d'attente.")

    def ajouter_personne_prioritaire(self, nom):
        self.file_prioritaire.append(nom)
        print(f"{nom} a été ajouté en tant que personne prioritaire.")

    def supprimer_personne_de_attente(self):
        if self.file_prioritaire:
            personne_supprimee = self.file_prioritaire.pop(0)
            print(f"{personne_supprimee} a été supprimé de la file
d'attente prioritaire.")
        elif self.file_attente:
            personne_supprimee = self.file_attente.pop(0)
            print(f"{personne_supprimee} a été supprimé de la file
d'attente.")
        else:
            print("La file d'attente est vide.")

```

```

from reservation import SalleCinema

def afficher_menu():
    print("\nMenu principal :")
    print("1. Réserver une place")
    print("2. Afficher les places réservées")
    print("3. Afficher le nombre de places disponibles")
    print("4. Filtrer les réservations par personne")
    print("5. Annuler une réservation")
    print("6. Réserver une place spéciale")
    print("7. Quitter")

def main():
    salle = SalleCinema(capacite=50, places_speciales=5)

    while True:
        afficher_menu()
        choix = input("Entrez votre choix : ")

```

```
if choix == "1":
    nom = input("Entrez votre nom : ")
    place = int(input("Entrez le numéro de la place : "))
    salle.reserver_place(nom, place)
elif choix == "2":
    salle.afficher_places_reserves()
elif choix == "3":
    salle.nombre_places_disponibles()
elif choix == "4":
    nom = input("Entrez le nom de la personne : ")
    salle.filtrer_reservations_par_personne(nom)
elif choix == "5":
    nom = input("Entrez le nom de la personne dont vous
souhaitez annuler la réservation : ")
    salle.annuler_reservation(nom)
elif choix == "6":
    nom = input("Entrez votre nom : ")
    salle.reserver_place_speciale(nom)
elif choix == "7":
    print("Programme terminé.")
    break
else:
    print("Choix invalide. Veuillez entrer un nombre entre 1 et
7.")

if __name__ == "__main__":
    main()
```