class SongNode:

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

self.title = title self.artist = artist self.next = None class Playlist: def \_\_init\_\_(self): self.head = None def insert\_front(self, title, artist): new\_song = SongNode(title, artist) new\_song.next = self.head self.head = new\_song

print(f'Song "{title}" inserted at the front.') def insert\_back(self, title, artist): new\_song = SongNode(title, artist) if not self.head:

self.head = new\_song

else:

current = self.head while current.next: current = current.next current.next = new\_song

print(f'Song "{title}" inserted at the back.'); def insert\_after(self, after\_title, title, artist):

current = self.head while current:

if current.title == after\_title: new\_song = SongNode(title, artist) new\_song.next = current.next current.next = new\_song

print(f'Song "{title}" inserted after

"{after\_title}".')

return current = current.next

print(f'Song "{after\_title}" not found. Cannot insert "{title}".') def delete\_front(self): if not self.head:

print("Playlist is empty. Nothing to delete at front.")

return

removed = self.head self.head = self.head.next

print(f'Song "{removed.title}" removed from front.') def delete\_back(self): if not self.head:

print("Playlist is empty. Nothing to delete at back.")

return if not self.head.next: removed = self.head self.head = None

print(f'Song "{removed.title}" removed from back (only song).')

return current = self.head while current.next.next: current = current.next removed = current.next current.next = None

print(f'Song "{removed.title}" removed from back.') def delete\_middle(self, title):

current = self.head prev = None while current:

if current.title == title: if prev: prev.next = current.next

else:

self.head = current.next

print(f'Song "{title}" removed from playlist.') return prev = current current = current.next

print(f'Song "{title}" not found in playlist.')

def display\_playlist(self): if not self.head: print("Playlist is empty.") return current = self.head idx = 1 print("\nPlaylist:") while current:

print(f"{idx}. {current.title} - {current.artist}") current = current.next idx += 1

print()

def search\_song(self, title): current = self.head while current: if current.title == title:

print(f'Found "{title}" by {current.artist} in the playlist.')

return current = current.next print(f'Song "{title}" not

found in playlist.') def main(): playlist = Playlist() while True:

print("\n--- Playlist Menu ---") print("1. Insert song at front") print("2. Insert song at back") print("3. Insert song after a song") print("4. Delete song from front") print("5. Delete song from back") print("6. Delete a specific song by title") print("7. Display playlist") print("8. Search song by title") print("9. Exit") choice = input("Enter your choice (1-9): ") if choice == '1':

title = input("Enter song title: ") artist = input("Enter artist name: ") playlist.insert\_front(title, artist) elif choice == '2':

title = input("Enter song title: ") artist = input("Enter artist name: ") playlist.insert\_back(title, artist) elif choice == '3':

after\_title = input("Enter the title of the song to insert after: ") title = input("Enter new song title: ") artist = input("Enter artist name: ")

playlist.insert\_after(after\_title, title, artist) elif choice == '4':

playlist.delete\_front() elif choice == '5': playlist.delete\_back() elif choice == '6':

title = input("Enter the title of the song to delete: ")

playlist.delete\_middle(title) elif choice == '7':

playlist.display\_playlist() elif choice == '8':

title = input("Enter the title of the song to search: ") playlist.search\_song(title) elif choice == '9':

print("Exiting... Goodbye!") break else:

print("Invalid choice! Please enter a number between 1 and 9.") if \_\_name\_\_ == "\_\_main\_\_":

main()

