

# 07-Java\_Assignment

## 🚺 ArrayList - Online Music Playlist 🎵

Scenario: Create a Music Playlist where users can add, remove, display, and shuffle songs.

#### Steps:

- Use an ArrayList<String> to store song names.
- Provide options:
  - Add a song
  - Remove a song
  - ✓ Display all songs
  - Shuffle songs (Use Collections.shuffle())

### Example Output:

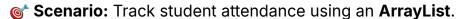
- 1. Add Song
- 2. Remove Song
- 3. Display Playlist
- 4. Shuffle Playlist

Enter choice: 1

Enter song name: Shape of You

Song added to playlist!

## 🔼 ArrayList - Student Attendance System 🎓



#### Steps:

- Store **student names** in an ArrayList<String>.
- Provide options:

- Mark student **Present** (Add name)
- ✓ Mark student Absent (Remove name)
- Display Present Students

### Example Output:

- 1. Mark Present
- 2. Mark Absent
- 3. Display Present Students

Enter choice: 1

# LinkedList - Browser History 🌐

**Scenario:** Simulate a browser history using a LinkedList.

### Steps:

- Use a LinkedList<String> to store visited websites.
- Provide options:
  - ✓ Visit a website (Add to list)
  - **▼** Go **back** (Remove last visited website)
  - Display browsing history

### Example Output:

- 1. Visit Website
- 2. Go Back
- 3. Show History

Enter choice: 1

Enter Website: youtube.com

youtube.com added to history!

# LinkedList - Task Manager

**Scenario:** Implement a **Task Manager** where users can **add, remove, and complete tasks**.

07-Java\_Assignment 2

#### Steps:

- Use a LinkedList<String> to store tasks.
- Provide options:
  - Add Task
  - ▼ Remove Task
  - ✓ Mark Task as Completed
  - ✓ View Pending Tasks

### Example Output:

- 1. Add Task
- 2. Remove Task
- 3. Mark Task Completed
- 4. View Pending Tasks

Enter choice: 1

Enter Task: Finish Java Assignment

Task added!

### Submission Guidelines:

- √ Implement each program separately & test it.
- ✓ Use proper method names & follow coding best practices.
- ✓ Make sure user input handling is done properly.

07-Java\_Assignment 3