```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace MP1
{
    interface ITeam
    {
        void AddPlayer(Player player);
        void RemovePlayer(int playerId);
        Player GetPlayerById(int playerId);
        List<Player> GetPlayersByName(string playerName);
        List<Player> GetAllPlayers();
    }

    // Player class
    class Player
    {
        public int PlayerId { get; set; }
        public string Name { get; set; }
        public int Age { get; set; }
    }

    // OneDayTeam class implementing ITeam interface
    class OneDayTeam : ITeam
    {
        private List<Player> players = new List<Player>();

        public void AddPlayer(Player player)
        {
            if (players.Count < 11)
            {
                players.Add(player);
                Console.WriteLine($"Player {player.Name} added to the team.");
            }
            else
            {
                Console.WriteLine("Cannot add more than 11 players to the team.");
            }
        }

        public void RemovePlayer(int playerId)
        {
            Player playerToRemove = players.FirstOrDefault(p => p.PlayerId ==
playerId);
            if (playerToRemove != null)
            {
                players.Remove(playerToRemove);
                Console.WriteLine($"Player {playerToRemove.Name} removed from the
team.");
            }
            else
            {
                Console.WriteLine("Player not found in the team.");
            }
        }
```

```
        public Player GetPlayerById(int playerId)
        {
            return players.FirstOrDefault(p => p.PlayerId == playerId);
        }

        public List<Player> GetPlayersByName(string playerName)
        {
            return players.Where(p => p.Name.Equals(playerName,
StringComparison.OrdinalIgnoreCase)).ToList();
        }

        public List<Player> GetAllPlayers()
        {
            return players;
        }
    }

    // Program class with the Main method
    class Program
    {
        static void Main()
        {
            OneDayTeam cricketTeam = new OneDayTeam();

            while (true)
            {
                Console.WriteLine("Enter 1: To add player | 2: To remove player by
id | 3: Get player by id | 4: Get player by name | 5: Get all players | 0: Exit");
                int choice;

                if (int.TryParse(Console.ReadLine(), out choice))
                {
                    switch (choice)
                    {
                        case 1:
                            // Adding players
                            Console.WriteLine("Enter player details: PlayerId, Name,
Age (comma-separated)");
                            string[] playerDetails = Console.ReadLine().Split(',');
                            if (playerDetails.Length == 3 &&
int.TryParse(playerDetails[0], out int playerId) && int.TryParse(playerDetails[2],
out int age))
                            {
                                cricketTeam.AddPlayer(new Player { PlayerId =
playerId, Name = playerDetails[1], Age = age });
                            }
                            else
                            {
                                Console.WriteLine("Invalid input. Please enter valid
details.");
                            }
                            break;

                        case 2:
                            // Removing a player by Id
                            Console.WriteLine("Enter player Id to remove:");
```

```
                            if (int.TryParse(Console.ReadLine(), out int
playerIdToRemove))
                            {
                                cricketTeam.RemovePlayer(playerIdToRemove);
                            }
                            else
                            {
                                Console.WriteLine("Invalid input. Please enter a
valid player Id.");
                            }
                            break;

                        case 3:
                            // Getting player details by Id
                            Console.WriteLine("Enter player Id to get details:");
                            if (int.TryParse(Console.ReadLine(), out int
playerIdToGet))
                            {
                                Player playerById =
cricketTeam.GetPlayerById(playerIdToGet);
                                if (playerById != null)
                                    Console.WriteLine($"Player details by Id:
{playerById.PlayerId}, {playerById.Name}, {playerById.Age}");
                                else
                                    Console.WriteLine("Player not found.");
                            }
                            else
                            {
                                Console.WriteLine("Invalid input. Please enter a
valid player Id.");
                            }
                            break;

                        case 4:
                            // Getting players by name
                            Console.WriteLine("Enter player name to search:");
                            string playerNameToSearch = Console.ReadLine();
                            List<Player> playersByName =
cricketTeam.GetPlayersByName(playerNameToSearch);
                            if (playersByName.Any())
                            {
                                Console.WriteLine($"Players with the name
'{playerNameToSearch}':");
                                foreach (var player in playersByName)
                                {
                                    Console.WriteLine($"{player.PlayerId},
{player.Name}, {player.Age}");
                                }
                            }
                            else
                            {
                                Console.WriteLine("No players found with the given
name.");
                            }
                            break;

                        case 5:
                            // Getting all players
```

```
                            List<Player> allPlayers = cricketTeam.GetAllPlayers();
                            Console.WriteLine("All players in the team:");
                            foreach (var player in allPlayers)
                            {
                                Console.WriteLine($"{player.PlayerId},
{player.Name}, {player.Age}");
                            }
                            break;

                        case 0:
                            // Exit the program
                            Environment.Exit(0);
                            break;

                        default:
                            Console.WriteLine("Invalid choice. Please enter a valid
option.");
                            break;
                    }
                }
                else
                {
                    Console.WriteLine("Invalid input. Please enter a valid
option.");
                }
            }
        }
    }
}
```