# Screenshot of code

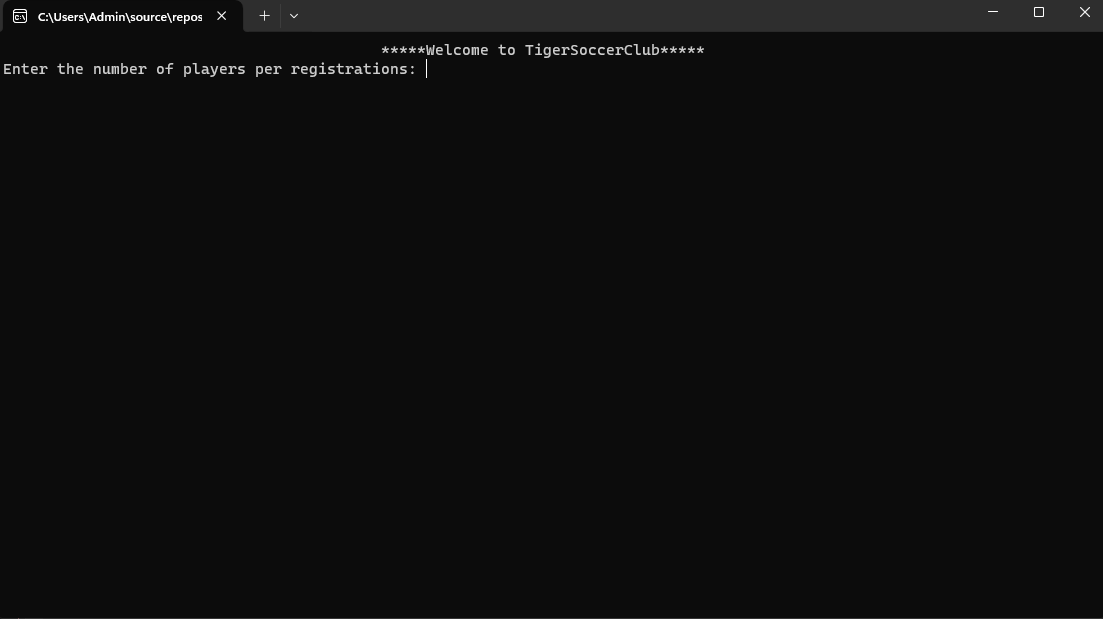


Fig: screenshot of tiger soccer club

# Problem Description

The programmer is attempting to develop a program for a soccer team that will manage player registrations, compute the overall fee based on the kind of registration (adult or kids), and determine whether or not the player want a jersey. The software seeks to gather information about each player, including name, registration type, and preferred jersey, as well as the number of players per registration, before computing the overall cost for each participant. At the conclusion of the program, there is a summary of applicants within the applications.

# Re-design of code

## Change-1

Trying to update the variables of code and make the code much more readable. Instead of a1, b2, c3 we can make use of the variable with naming conventions which indicate the basic purpose.

## Change-2

The second change will be to extract the common calculations and logic, which meanwhile reduces the duplication and makes easier to maintain.

## Change 3

Try to move the summary printing out of loop.

## Change-4

Try to improve the formatting of code such as the overall summary of table by aligning the columns properly.

# Re-designed code and output

using System;

namespace TigerSoccerClub

{

class Program

{

private static int initalamt\_kids = 150;

private static int initalamt\_adult = 230;

private static int jerseyamt = 100;

public Program(int players, string name, string registration, string jersey, double price)

{

Players = players;

Name = name;

Registration = registration;

Jersey = jersey;

Price = price;

}

public int Players { get; set; }

public string Name { get; set; }

public string Registration { get; set; }

public string Jersey { get; set; }

public double Price { get; set; }

static void Main(string[] args)

{

Console.Write("Enter the number of players per registrations: ");

int totalPlayers = Convert.ToInt32(Console.ReadLine());

if (totalPlayers > 4 || totalPlayers < 1)

{

Console.WriteLine("Invalid number, Please enter the registration number between 1 to 4");

}

else

{

Program[] playersArray = new Program[totalPlayers];

for (int i = 0; i < totalPlayers; i++)

{

GetPlayerDetails(out string name, out string registration, out string jersey);

double total = CalculateTotalPrice(registration, jersey);

PrintRegistrationDetails(name, registration, jersey, total);

playersArray[i] = new Program(totalPlayers, name, registration, jersey, total);

}

PrintSummary(playersArray);

}

}

static void GetPlayerDetails(out string name, out string registration, out string jersey)

{

Console.Write("Enter name: ");

name = Console.ReadLine();

Console.Write("Registration type: ");

registration = Console.ReadLine();

Console.Write("Enter Yes/No to indicate whether you want a jersey: ");

jersey = Console.ReadLine();

}

static double CalculateTotalPrice(string registration, string jersey)

{

double basePrice = (registration == "Kids") ? initalamt\_kids : initalamt\_adult;

double total = jersey.ToLower() == "yes" ? basePrice + jerseyamt : basePrice;

double discount = (total \* 5) / 100;

return total - discount;

}

static void PrintRegistrationDetails(string name, string registration, string jersey, double total)

{

Console.WriteLine($"Total price from {name} is: {total}\n");

}

static void PrintSummary(Program[] playersArray)

{

Console.WriteLine("Summary of Registrations");

Console.WriteLine("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

Console.WriteLine("Name\tType\tJersey\tTotal");

Console.WriteLine();

foreach (var player in playersArray)

{

Console.WriteLine($"{player.Name}\t{player.Registration}\t{player.Jersey}\t{player.Price}");

}

}

}

}

