Students:

This content is controlled by your instructor, and is not zyBooks content. Direct questions or concerns about this content to your instructor. If you have any technical issues with the zyLab submission system, use the **Trouble with lab** button at the bottom of the lab.

25.5 Q5 (10 marks)

Complete the provided skeleton C++ code as detailed in the tasks below which relate to a Sports Team information system. You are provided with the following:

- `main.cpp`: this contains code which you can use to develop your code. It will not be used in the unit tests.
- Base class:
    - `Team.h`: (header file) you will **write assessed C++ code in this file**
    - `Team.cpp`: contains the member function definitions and implementation details (this file is hidden)
- Derived class:
    - `SoccerTeam.h`: (header file) **you will write assessed C++ code goes here**
    - `SoccerTeam.cpp`: **you will write assessed C++ code in this file**
- `TeamList.h`: header file (read only)
- `TeamList.cpp`: **you will write assessed C++ code in this file**
- `utilities.h` - includes `escapeString` (see zylab 20.2)

Task 1

Given the base class `Team` (see file `Team.h`), complete the derived class `SoccerTeam` with the constructor:

```
SoccerTeam(std::string name = "No name", std::string coach = "No coach",
           int played = 0, int won = 0, int lost = 0, int drawn = 0);
```

- This constructor has 6 parameters: `name`, `coach`, `played`, `won`, `lost` and `drawn`.
- The parameter values should be assigned to the respective data members.

Task 2

Complete the `SoccerTeam` member function `std::string formatInfo()` which formats a `std::string` as follows and shown in th example below:

- all member fields are separated by `|` followed by a single space `" "`
- `team` and `coach` are left justified in a field width of 20 (ie exactly 20 spaces between each `|`)

- all other fields are left justified in a field width of 10 (ie exactly 10 spaces between each `|`).
- The overall length of the string is 87: the sum of each field + 7 separator characters `|`

```
| name               | coach              | played   | won      |
lost     | drawn    |
------------------------------------------------------------------
---------------------
|<----- 20 --------->|<----- 20 --------->|<-- 10 -->|<-- 10 -->|
<-- 10 -->|<-- 10 -->|
```

Example

For the following `SoccerTeam` object

```
SoccerTeam nufc("Newcastle United", "Eddie Howe", 32, 15, 12, 5);
nufc.formatInfo();
```

the method should return the following string:

```
"| Newcastle United   | Eddie Howe         | 32       | 15
| 12       | 5        |"
```

Task 3

Given `TeamList.h` complete the member function

```
 std::string formatTeamList();
```

Details are:

- it has no input parameters
- it returns a formatted string of the objects in `TeamList` (class data member) which can be either:
  - `Team` base class objects
  - `SoccerTeam` derived class objects
- the `Team` objects should be formatted using the base class member function `formatInfo()`
- the `SoccerTeam` objects should be formatted using its member function `formatInfo()` (i.e. override the base class function).
- the

NOTE

- You will need to use polymorphism and virtual functions to allow for function overriding
- You will need to make changes in the following files:
  - `Team.h`
  - `SoccerTeam.h`

Example for the following:

```
TeamList teamList;
teamList.addTeam(&nufc);
teamList.addTeam(&hawks);
teamList.addTeam(&spurs);
teamList.addTeam(&demons);
```

the following:

```
teamList.formatTeamList()
```

should return a formatted table, which when output is as follows:

```
| Name               | Coach               | Played   | Won      |
Lost     | Drawn    |
==================================================================
| Newcastle United   | Eddie Howe          | 32       | 15       |
12       | 5        |
------------------------------------------------------------------
----------------------
| Hawthorn           | Sam Mitchell        | 15
|
------------------------------------------------------------------
----------------------
| Tottenham Hotspurs | Ange Postecoglou    | 32       | 18       |
8        | 6        |
------------------------------------------------------------------
----------------------
| Melbourne          | Simon Goodwin       | 5
|
------------------------------------------------------------------
----------------------
```

the actual string that is returned (printed using `escapeString`), should be:

```
"| Name | Coach | Played | Won | Lost | Drawn
|\n==================================================================
Newcastle United | Eddie Howe | 32 | 15 | 12 | 5 |\n---------------
-------------------------------------------------------------------
-----\n| Hawthorn | Sam Mitchell | 15 |\n--------------------------
-------------------------------------------------------------\n|
Tottenham Hotspurs | Ange Postecoglou | 32 | 18 | 8 | 6 |\n--------
-------------------------------------------------------------------
------------\n| Melbourne | Simon Goodwin | 5 |\n------------------
```

```
---------------------------------------------------------------------
--\n"
```

NOTE the actual string that is returned by the Base Class `Team` function `formatInfo` will need padding at the end to match the required format of the table. For example, for the folllowing:

```
Team hawks("Hawthorn", "Sam Mitchell", 15);
std::string test = hawks.formatInfo();
std::cout << escapeString(test) << "\n";
```

the string returned (printed using `escapeString`):

```
"| Hawthorn            | Sam Mitchell        | 15        "
```

But this needs padding at the right hand end and the addition of `'|'` (so that the total length of the string is 87 characters)

671842.4329690.qx3zqy7

LAB ACTIVITY 25.5.1: Q5 (10 marks) 0 / 10

Current file: **main.cpp** ▾ **Load default template...**

```
#include <iostream>
#include "Team.h"
#include "Team.cpp" // You can use this class (implementation is hidden)
#include "SoccerTeam.h"
#include "TeamList.h"
#include "utilities.h" // Includes escapeString(std::string) for use to d
#include "utilities.cpp"

int main()
{
    //  /***************************************************************
    //    * This code is provided as a starting point for development and
    //    * Please modify the code below as you develop your program.
    //    * ***************************************************************

    SoccerTeam nufc("Newcastle United", "Eddie Howe", 32, 15, 12, 5);
    SoccerTeam spurs("Tottenham Hotspurs", "Ange Postecoglou", 32, 18, 8,
    Team hawks("Hawthorn", "Sam Mitchell", 15);
```

Develop mode **Submit mode**

Run your program as often as you'd like, before submitting for grading. Below, type any needed input values in the first box, then click **Run program** and observe the program's output in the second box.

Enter program input (optional)

If your code requires input values, provide them here.

Run program

Input (from above) → **main.cpp** (Your program) →

Program output displayed here

Coding trail of your work What is this?

```
History of your effort will appear here once you begin
working on this zyLab.
```

Trouble with lab?