



# North South University

Department of Electrical and Computer Engineering

CSE 215L (Programming Language II Lab)

Lab 6: Classes and Objects (II)

## Objective:

- Master using class objects as parameter
- Master using an array of objects

## Objects as Parameter in methods:

We have been using parameters of certain data types in the methods for a long time. However, now you know that class objects can also be used as Parameters.

All you need to do is insert an object as a method parameter so that the object can be used inside the method for various purposes.

For example, for the Circle class, we create a method called `compareGreaterRadius(Circle c)`, where we compare the radii of the two circles and return True if the current Circle object from where the method has been invoked has a greater radius than the radius of the Circle object parameter. The invoking of the method is given as below:

```
c1.compareGreaterRadius(c2)
```

Where `c1` is the current circle object from where the method has been called, and `c2` is the object used as a parameter.

## An array of Objects:

Do you know that it is possible to create an array of class objects, it can be declared in the same way you have been declaring an array for other data types for so long. For example, declaring an array of Circle objects is given as follows:

```
Circle [] circleArray = new Circle[10]
```

Here, we declare an array of Circles with size 10, meaning there can be 10 circles in the array. However, you need to initialize the circle array at first by using a for loop like this one:

```
for (int i = 0; i < circleArray.length ; i++){  
    circleArray[i] = new Circle();}
```

## Classwork:

1. Create a class named 'Footballer', which has the following data fields  
: a. name      b. team      c. position      d. matches      e. goals      f. assists
2. The footballer class should ONLY have a default constructor, having a constructor with parameters is declared ILLEGAL.
3. The footballer should have to get and set methods for name, team, and position.
4. Create methods to increase the number of matches, goals, and assists.
  - a. The number of matches will increase by 1; no parameters
  - b. The number of goals will increase by the provided parameters
  - c. The provided parameters will increase the number of assists.
5. Create methods to compare the number of goals and assists with another footballer where the current footballer is being compared to another footballer. (Hint: the parameter to be used is another)
  - a. If the current footballer has more goals and assists than the other player, then the method will print "<name of current footballer> has more goals and assists than <name of another footballer>."
  - b. If the current footballer has more goals but fewer or equal assists than the other player, then the method will print "<name of current footballer> has more goals but fewer or equal assists than <name of another footballer>."
  - c. If the current footballer has more assists but fewer or equal goals than the other player, then the method will print "<name of current footballer> has more assists but fewer or equal goals than <name of another footballer>."
  - d. Otherwise, print "<name of current footballer> has fewer goals and assists than <name of another footballer>"
6. Then, create two Footballers with the following information given below

	Player 1	Player 2
Name	Kylian Mbappe	Erling Haaland
Team	PSG	Manchester City
Position	Left-Winger	Center-Forward

So far, they have played 3 matches with the following statistics, given the goals and assists as user input, and should strictly follow this pattern.

	Kylian Mbappe		Erling Haaland	
	Goals	Assists	Goals	Assists
Match 1	2	0	2	1
Match 2	3	0	2	1
Match 3	2	2	2	2

Then, you will compare their goals and assist using the comparing method you created earlier.