# Lab 7: Designing a sports league program

/\*\*

\* @author Monika Szucs

\* @author Partner here

\* @version 1.0

\*/

## Theme

**Hockey**

Consider what we learned on:

* RDD
  + Hockey team has its own well-defined role
  + The goal of the larger program is realized when each of the smaller parts fulfills its purpose
* Cohesion
  + Refers to the number of diversity of tasks that a single unit is responsible for. We aim for HIGH COHESION
* Coupling
  + Avoid Coupling – we aim for loose coupling
  + Links between separate units of a program
  + When for example two players depend on each other so closely that neither can function on their own so we lose cohesion and the role gets confused
* Duplication
  + When the same or very similar code sequence appears more than once in the same class or in different classes within the same program

## Questions to consider:

For players:

1. What attributes does a player have?

* firstName
* lastName
* baseSalary
* jerseyNumber

1. What fields would be required?
2. What constructors are required to create players?

* Like the other lab that had 3, 4, or 5 parameters with different conditions inside the constructor – just do what I think Is best and describe it here

1. What validations are required?

Check this is String firstName

* Shouldn’t be null or empty

Check this is String lastName

* Shouldn’t be null or empty

Check this is double baseSalary

* Can check to see if the salary is between a range

Check this is int jerseyNumber

* Shouldn’t be a negative number

1. What functions does a player perform?

For teams:

1. Same as above pertaining to teams

For the league:

1. Same as above for the league

## Additional things to consider

1. What are the relationships between the three classes? How do teams track players and how does the league track teams?
2. Is the design reusable?

* If I add more team players then it does the same checks

1. Is the design extendable? – ask teacher
2. Clarity. It should be clear to the Customer what you have designed. Remember, they won’t be programmers so you have to be reasonably detailed in your descriptions and explanations.