**Important Instructions:**

* **At the end of assignment, participants will able to implement Bean Auto wiring and AOP.**
* **Your code will be graded both on correctness and efficiency.**
* **Use comments in your code that explains your assumptions and design decisions.**
* **You need to submit your assignment solution by end of the day.**
* **Before submitting your assignment makes sure it as per the given requirement.**
* **You need to submit your assignment on mercury on the respective folder.**
* **Your folder name will be your employeeId\_Day\_Assignment. (E.G 105678\_Day1\_Assignment).**

**Q1.**  **Make a Spring Application as per the below given specification:**

i) Design a class Course having following properties

cId int

name String

fee float

ii) Override toString() to display Course information.

iii) Design a class Student having following properties

sId int

name String

Set<Course> course;

iv) Inside student class declare a method printInfo() which will print Student and Course

information.

V) Declare a class MainApp having main method and bootstrap the Spring application using

ApplicationContext .

**Note : Use Constructor based dependency injection**

**Q2.**  **Make a Spring Application as per the below given specification:**

i) Design a class Club having following properties

cId int

name String

owner String

ii) Override toString() to display Club information .

iii) Design a class Player having following properties

pId int

name String

pay float

iv) Declare a method printInfo() inside Player class to display all the information.

**v) Declare a class MainApp having main method and bootstrap the Spring application using ApplicationContext**

**Note : Use all four types bean autowiring mode to resolve the dependency**

* **byName**
* **byType**
* **constructor and**
* **autodetect**

**Q3.**  **Make a Spring Application as per the below given specification:**

i) Design an Interface EmployeeService having the method

public void getEmployeebyName(String name);

ii) Design a class EmployeeServiceImpl which implements EmployeeService interface

implement the getEmployeebyName(String name) method this will print the name of the

employee. If name is null then this method throw NullPointerException.

iii) Design a class EmployeeAspect having following members

* Declare a pointcut named as allMethod()
* Declare a method log() and associate allMethod pointcut to this method the log method print the time taken by the getEmployeebyName(String name) mthod to execute. Block the execution of log method for two seconds.
* Declare a method afterThrowing() which apply on getEmployeebyName(String name) if this method throw an exception print the message employee name can’t be null inside afterThrowing() method

**Note : Use AOP**