Java Exercise

Given Date – 06/09/2016

Finish Date – on or before 06/12/2016

Good old POJO – poor Employee

1. Create a class called Employee that holds the following info.
    a. id – integer
    b. fname – String
    c. lname – String
    d. dob – Date (java.util)
    e. sex – String
    f. address – Address type (which has line 1, line 2, zipcode, state, country all are Strings)
2. Have relevant constructors – [*one twist…. the id of the employee never allowed to be passed as argument from the constructor. Instead it is assigned from an object id pool. Assume that you have an id pool 1 with 1 to 10 values…. use the ide from there, if it is not already assigned….*]

3. Have relevant getters/setters for getting/setting the fields data
4. Follow the rules of access specifiers
5. Make sure once an object of Employee is created, it will never change – take all good measures…. ;-)
6. BTW, what is this called?

Sachin Tendulkar – God of Cricket's "tons"

Problem Description: You are given the task (golden opportunity ☺) of keeping statistics of the Cricket God's centuries information. (BTW, centuries are referred as "tons" sometimes). Whenever a century is made, the following info needs to be captured:

1. Date
2. Venue (the stadium name and country)
3. Against (WI, AUS etc.)
4. Type of match: (ODI/Test/T20)
5. Runs
6. Out/Not-out
7. No. of 4's
8. No. of 6's

[Hint: if Sachin made 51 centuries in tests plus 123 centuries in ODI, your database (or storage) would have those many records – Consider you will have it in memory]

Now, you need to provide public API to the customers like “mee TV, eeTV, ESPN” etc. to provide this info

- How many total centuries the God made so far in all forms of matches?
- What is the highest century stats:
    - When?
    - Against Whom?
    - How many 4s and 6s in that?
    - Out/Not-out?
    - Type of match
- Total number of runs made in all matches
    - Among them how many times outs
    - Not-outs
- Average record
- of (each not-out reduces the average… if out of 10 matches played, only 5 times out and remaining not-outs, then sum the total runs and divide by 5, not by 10)
    - ODIs
    - Test matches
    - T20s

Model this API such that:

- whenever a new century is made, a method on this API needs to be called to register that century
- the stats info comes from this public API
- you can use all the concepts of class hierarchies, or interface/abstractions, singletons/factories etc.
- **there may be a future need to capture the same info for Virat Kohli. So think about the future…. when you design ;-)….**

<u>**Small exercise on Collections**</u>

- assume u have words in 2 different txt files
- read 2 text files from disk (hint: use nio package)
- store the contents in 2 collections
- sort the 2 collections in ascending order
- find out the count of each word from file1 (collection1) in file2