**Instructions – Final Exam Coding Section**

**Overview:** A local gym wants to test out a simple prototype program for importing a sample of their data about members and trainers and processing it. You are given two excel files, one with data about gym members, the other with data about gym trainers. Gym members can be standard members, or premium members. The only difference is that premium members are assigned a gym trainer. The gym’s policy is that gym trainers are always the same gender as the premium member they train. You’ll need to import the data into python, create classes for standard members, premium members, and trainers, and display some information about them.

Remember that partial credit is given. Do as much as you can.

**Libraries Required:**

* import random
* choose one:
  + import pandas as pd
  + import openpyxl

**External Files Required:**

* gym\_members.xlsx (download from learning suite)
  + includes name, gender, number of days they’ve been a member, number of times they’ve visited the gym, their favorite workout, and the type of member they are (Standard or Premium)
* gym\_trainers.xlsx (download from learning suite)
  + includes name and gender.

**Classes Required:**

* StandardMemeber
  + Instance variables:
    - name, gender, num\_days\_member, num\_gym\_visits, favorite\_workout, membership\_type
  + Methods:
    - \_\_init\_\_ (the constructor)
    - recommended\_action
      * prints out a recommended action based on how often they’ve gone to the gym.
* PremiumMember (must inherit from StandardMember)
  + Instance variables:
    - Everything from StandardMember through the use of super()
    - Trainer (will store a trainer object, but by default it should be equal to None)
  + Methods:
    - \_\_init\_\_(the contructor)
    - recommended\_action
      * prints out a recommended action based on how often they’ve gone to the gym, and how their trainer feels about it.
* Trainer
  + Instance variables:
    - name, gender
  + Methods:
    - \_\_init\_\_ (the constructor)

**Logical Flow:**

There are many ways to do this. All that matters is that your code implements the requirements. The exact order, what libraries you use, the specific code, etc. can all vary. The only exception would be doing something in a very, very obviously inferior way (such as hard coding/manually typing in all the information from the files instead of importing it using pandas/openpyxl, etc.)

1. Import the gym\_members.xslx file (you can use relative or absolute paths) and have each row in the file become either a StandardMember object or a PremiumMember object, based on whether it says “Standard” or “Premium” under the membership\_type column in the Excel file.
   1. Every column in the Excel file represents what should become an instance variable for the StandardMember or PremiumMember objects.
   2. PremiumMembers must inherit from StandardMembers (e.g. PremiumMember is the child class of StandardMember)
   3. Every member (whether standard or premium) should be stored in a list.
2. Import the gym\_trainers.xlsx file (you can use relative or absolute paths), and have each row in the file become a Trainer object.
   1. Every column in the excel file represents what should become an instance variable for the Trainer objects.
   2. Every trainer should be stored in a list.
3. All PremiumMembers should have a trainer randomly assigned to them of the same gender. “trainer” is an instance variable that only PremiumMembers have, and it should hold a Trainer object.
   1. For example, Bob Johnson has a gender of “M” and is a premium member, so when he gets a trainer assigned, it should randomly be one of the 3 trainers that also have “M” as their gender.
4. Every member (whether standard or premium) should have the “recommended\_action()” method run on them.
   1. For StandardMembers:
      1. It will calculate the ratio of how often the member has visited the gym, divided by how many days they’ve been a member (num\_gym\_visits / num\_days\_member). A value of 1 means they’ve gone every single day since they joined, value of .5 means they’ve gone half of the days since joining, etc.
      2. If the ratio is .5 or above, display the message “*Recommended Action for name (gender): Keep up the routine!*”
      3. If the ratio is .3 or above, display the message “*Recommended Action for name (gender): Slightly increase visits.”*
      4. If the ratio is any lower, display the message “*Recommended Action for name (gender): Increase your frequency*.”
   2. For PremiumMembers:
      1. Do the exact same as above, but you’ll include some extra text with how their trainer feels:
      2. If the ratio is .5 or above, display the message “*Recommended Action for name (gender): Keep up the routine! trainer name (trainer gender) is proud!*”
      3. If the ratio is .3 or above, display the message “*Recommended Action for name (gender): Slightly increase visits. Trainer name (trainer gender) is encouraging you!”*
      4. If the ratio is any lower, display the message “*Recommended Action for name (gender): Increase your frequency. Trainer name (trainer gender) still believes in you!”*

Upload just the python file to Learning Suite.

**Example Output:**

Recommended Action for Alice Smith (F): Slightly increase visits.

Recommended Action for Bob Johnson (M): Keep up the routine! Jordan Kim (M) is proud!

Recommended Action for Charlie Williams (M): Slightly increase visits.

Recommended Action for Diana Brown (F): Keep up the routine! Sam Lee (F) is proud!

Recommended Action for Evan Davis (M): Increase your frequency.

Recommended Action for Fiona Taylor (F): Keep up the routine! Taylor Patel (F) is proud!

Recommended Action for George Moore (M): Keep up the routine!

Recommended Action for Hannah Jackson (F): Keep up the routine! Sam Lee (F) is proud!

Recommended Action for Ian Martin (M): Keep up the routine!

Recommended Action for Julia Lee (F): Increase your frequency. Taylor Patel (F) still believes in you!

Note: since trainers are randomly assigned, results will have some variation.

**Requirements:**

|  |  |
| --- | --- |
| **Requirement** | **Sub Requirements** |
| Imports gym\_members.xlsx | * + - Each row should become an object. |
| Imports gym\_trainers.xlsx | * + - Each row should become an object. |
| Includes StandardMember class | * + - Includes instance variables: name, gender, num\_days\_member, num\_gym\_visits, favorite\_workout, membership\_type     - Includes constructor method. |
| Includes PremiumMember class | * + - Includes instance variables: everything from StandardMember by using super(), trainer     - Includes constructor method. |
| Includes Trainer class | * Includes instance variables: name, gender. * Includes constructor method. |
| All members stored in a list and all trainers stored in a list |  |
| Premium members have trainer object assigned to them | * Entire trainer object should be stored in the PremiumMember trainer instance variable. * Trainer’s gender should match the PremiumMember’s gender. |
| StandardMember and PremiumMember class has recommended\_action() method. | * Should properly calculate ratio of gym visits to days since becoming member. * Should print out the proper message based on the ratio and include name and gender. * Should include name and gender of trainer for PremiumMembers. * Should be run on all members. |
| includes useful comments | * Should include a name and description at the top. * Should also include comments throughout. |