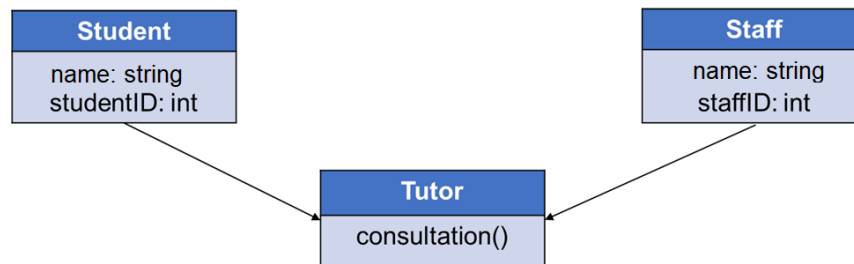


Module 6 – Exercises

1. Class Inheritance & Function Overriding

Create a hierarchy of classes as below, with attributes and methods are specified in each class.



*Note: for each Tutor object, there are **two versions** of **name** attributes inherited from Student and Staff classes. We should use only one version consistently, e.g. from the Student class by **Student::name**. As a result, you can access name of a tutor with **tutor.Student::name***

*For **consultation()** method, you only need to print out a message, e.g. “Doing consultation !”*

- Create **constructors** and **destructors** for all classes, both of which announce themselves to console output. *Verify that for a Tutor object, all three constructors and destructors are automatically called. Explain the order in which the calls are made.*
- Write a method namely **setName()** for the class Tutor which accepts a string to set the Tutor's name.
- Write a method namely **toString()** for all classes, which return all information as a string. Write two external functions:
 - showInfoStudent(Student &stu):** print out all information of a Student or Tutor
 - showInfoStaff(Staff &staff):** print out all information of a Staff or Tutor

Test all the methods written in question b and c in main().

2. Vector of Objects & Data Saving/Loading

- Reuse the class **Student** in question 1, write another class namely **SchoolSystem** with an attribute namely **students** which is a *vector of Student objects*. Write two methods for this class as below
 - addStudent():** prompt user to input name and student ID for a student and add to **students** list.
 - removeStudent():** prompt user to input student ID and remove the corresponding student from **students** list.
- In main() function, provide a menu as below for user. Save all information of students into a file namely **students.dat** at the end, and reload all data back at the beginning.

School Management System
1. View all students

2. Add Student

3. Remove Student

4. Exit

Enter Your Choice:.....

3. Copy Constructor and Copy Assignment Operator

Duplicate the program of the **previous week's question 2**, and write **Copy Constructor** and **Copy Assignment Operator** for the class Book. Test them in main().