Creative Software Programming Assignment#1 (week-11)

Every assignment will be announced on **Thursday** and should be sumitted by next **Thursday**.

In this week **Handed out will be Nov 12, 2020, Due Nov 19, 2020**

- College classes have a name (std::string).
- Building class has name (std::string), id (size_t), college (College&), and number of rooms (size_t).
    - People can enter the building. The admission follows the following rules.
        - There can only be one Employee per building.
        - There can be only one Professor in each room. If the number of rooms is 4, professors can enter up to 4 people.
        - Students can only enter the college building to which they belong.
        - People enter a random room when entering.
    - People can leave the building.
- There are three types of user classes: Student, Professor, and Employee.
    - User class is initialized with name (std::string), id (size_t), and college (College&).
        - Professor can call attendance. The id of all students in the professor's room are printed.
        - Employee can clean the building. Everyone except employee self must leave the building.

The input is:

- The number of colleges N is given on the first line.
- College names are given from next N lines.
- The next line is given the number of buildings B.
- Buildings are given next B lines in the form of `bulding_name building_id college_name number_of_rooms`.
- The number of people U is given on the next line.
- The next U lines, users are given in the form of `{student|professor|employee} user_name user_id college_name`.
- The command number C is given on the next line. The commands are as follows:
    - `enter user_ID building_ID`
        - print room number (starts with 1), if not enter then print 0
    - `exit user_ID building_ID`
    - `total building_ID`
        - print size_t
    - `clean user_ID`
    - `attendance user_ID`
        - print student's id
    - `where user_ID`

        - Displays the building_name where the user with user_ID currently exists. If not, output 0.
    - `all building_ID`
        - Prints the name of all users in the building.

The following errors are handled as follows.

- "Name duplicated error" when a college of the same name is established
- "Id duplicated error" when a building with the same ID is established or a user with the same ID is created
- "College not exist error" when a building or user of a college without a name is created
- "Not enter error" if the user is unable to enter the building for any reason
- "No students error" if the professor is not in the room or there are no students to attend
- "Nothing to clean error" when an employee is performing cleaning while not in the building
- "Permission error" in case of user behavior that does not match the permission

Issues

- All output is line by line. When calling attendance, each student id must be printed, one per line.
- You can create new functions, but you cannot declare additional member variables in class
- In the main function, do not create containers other than `colleges`, `buildings`, and `users`.
- Selecting arbitrary elements in a container can be done using `util::select_randomly` in utility.h.

```
std::vector<User*> v;
...
User* random_user = util::select_randomly(v.begin(), v.end());
```