Tasks - 3

Check the following UML representation for the `Shape` hierarchy:
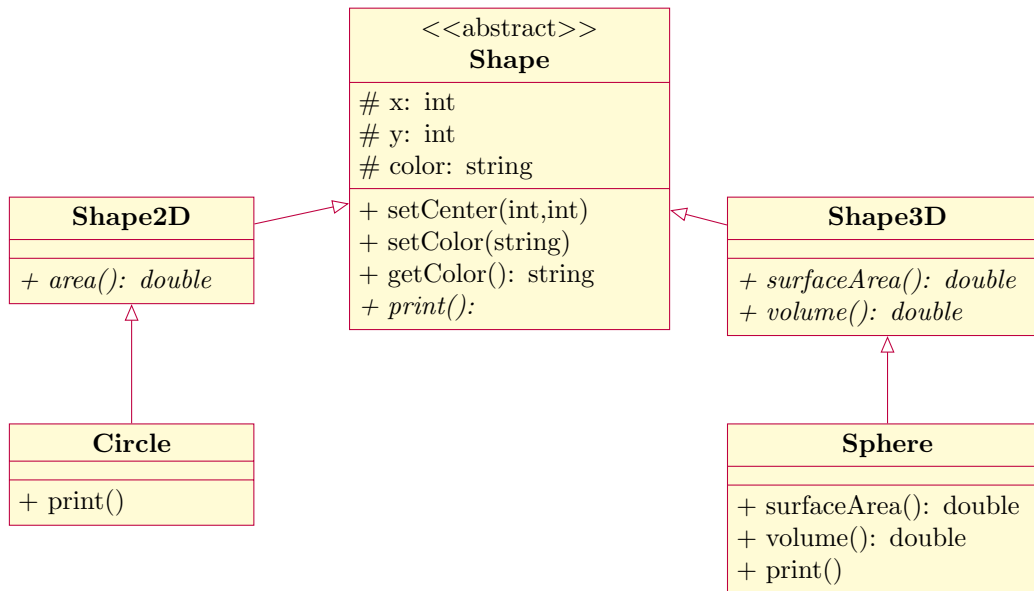
In UML (*http://www.uml-diagrams.org/class-reference.html*) the △ represents specialization, *Italic* methods are virtual, the symbol + represents public visibility and # represents protected visibility.

Implement this hierarchy in C++ and write a main program that exemplifies its use:

- add private data members to any of the classes if necessary;

- implement each function to output its name (e.g., `"Circle::print"`);

- create a client function (e.g. `Display` that uses the `print` function) that receives a reference to a `Shape` class (`Shape&`) and shows its use receiving instances of the concrete classes;

- write an user interaction to select concrete Shapes at run-time and and passes the created object to the `Display` function.