1. The keyword friend grants access private data or methods not in the current scope. After you do declare friend in a class, anything that follows after friend will be able to have access to private, and protected class members in the class that is calling friend.
2. Using namespace std has a lot of predefined identifiers. There might be a time where a class is made using the same predefined identifiers, and thus a name collision may occur; meaning that namespace std might have a same function name as another class, further complications can occur if the same functions accept the same parameters.
3. Ju
   1. Returning a reference to a class member: mark the method as const, and dereference itself. const angle& getAngle(){return angle;} this returns an object that you cannot modify, but it is a member of whatever class the method is in.
   2. Returning a copy: returns a new object. Example:

Angle superman::getaAgle() const {

Angle angle();

return angle;

} The person who called this method will store the angle and use it however they want.

* 1. Returning an object created on the heap: always want to return a pointer to heap data to ensure proper cleanup (deleting heap data). Create objects on heap using the “new” keyword. The object outlives the method because data was allocated on the heap.

\*Angle superman::getaAgle() const {

Angle\* angle = new Angle();

return angle;

}

D. use an argument: passes an instance of an object though referencing it, that can be manipulated inside the method. The code below will set the angle that was passed in to 180.

Void superman:: getAngle(Angle& angle) const{

angle.set(180);

}

4. The mutable keyword makes whatever is after the keyword exempt from const.

5.

A. a class can create an object, Fish goldfish(true); goldfish is a variable name for the object type of fish.

B. a class can have a pointer that references it to another object. Fish \*goldfish; I suppose if you reference an object, the class does not contain the object, rather it contains a reference to an object, and not the actual object.

C. A class can store data on the heap, Angle\* angle = new Angle();

D. a class can contain objects in an array of pointers, or can be called a pointer to a pointer.

6. A singleton class is a class that can only have one instance, or no instances at any given time in a program. We can do this by making the constructor private, and using the static keyword in a public method to initialize the singleton class. Static in c++ means that it runs whatever follows the keyword static once, and will never run it again throughout the whole program.

7.program not compiling correctly. Working on: Attempting to save a reference of population in the fish object, due to the fact that the deconstructor does not take any parameters.

First i was trying to save a reference to an angle

Header file: private: Population \*population;

Fish::Fish(double directionDegree, Population &population) : direction(directionDegree)

this->population = \*population;

Couldn’t do that because i guess you cant save a pointer when you derferrenced something.