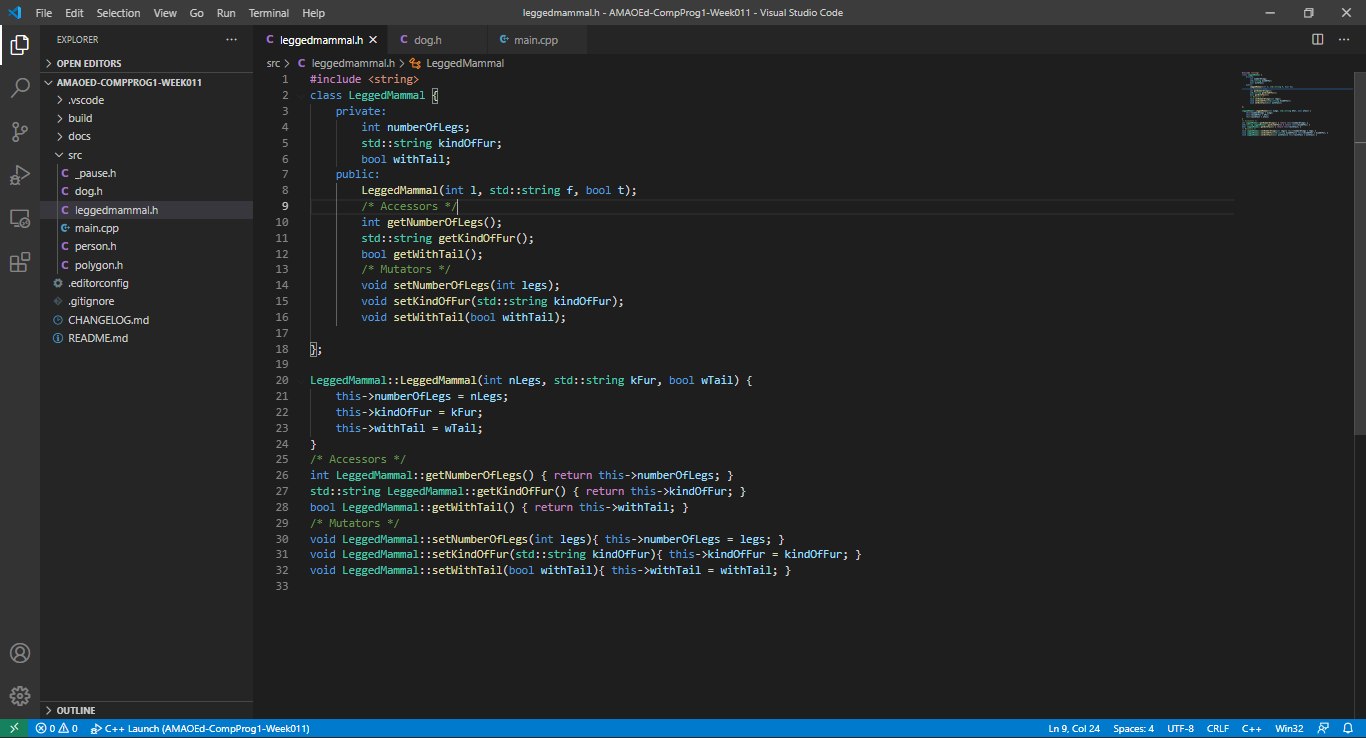
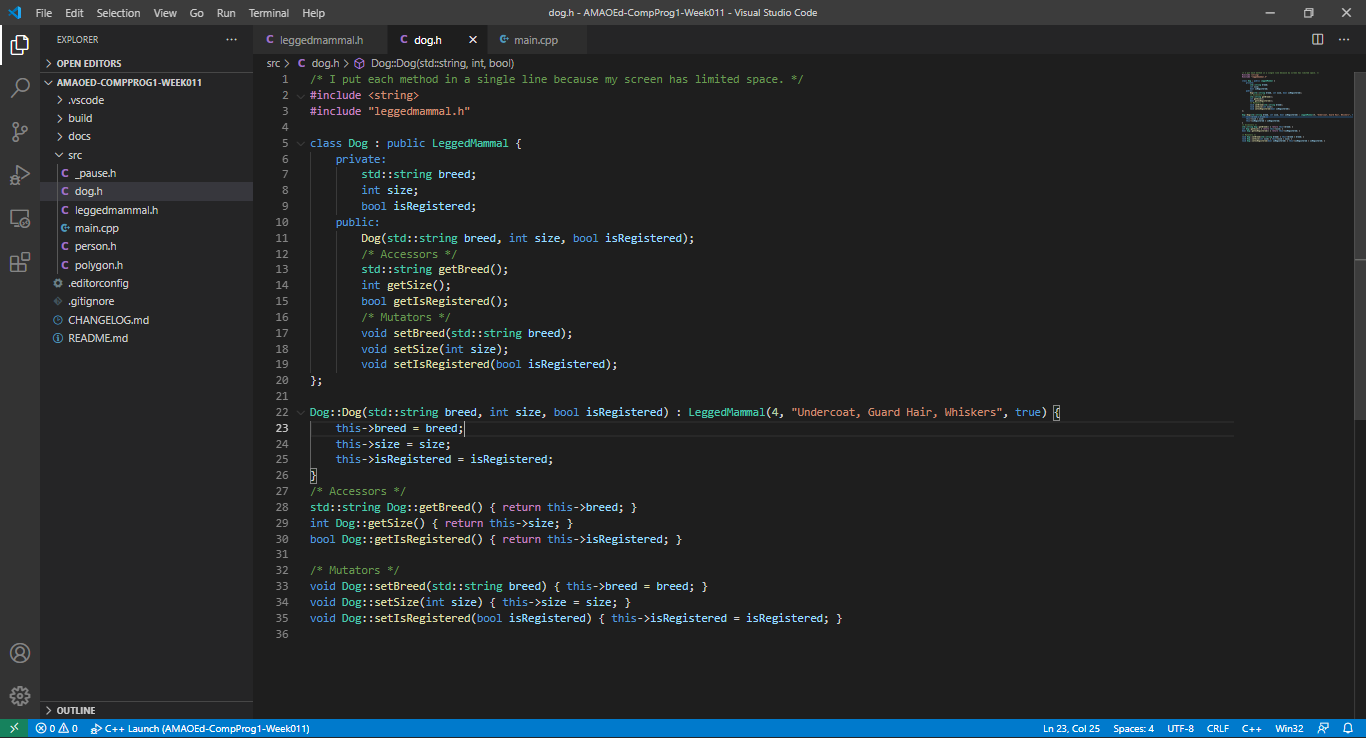
1. Write a class that extends the LeggedMammal class from the previous laboratory exercise. The class will represent a Dog. Consider the breed, size and is registered. Initialize all properties of the parent class in the new constructor. This time, promote the use of accessors and mutators for the new properties. Instantiate a Dog object in the main function and be able to set the values of the properties of the Dog object using the mutators. Display all the properties of the Dog object using the accessors.

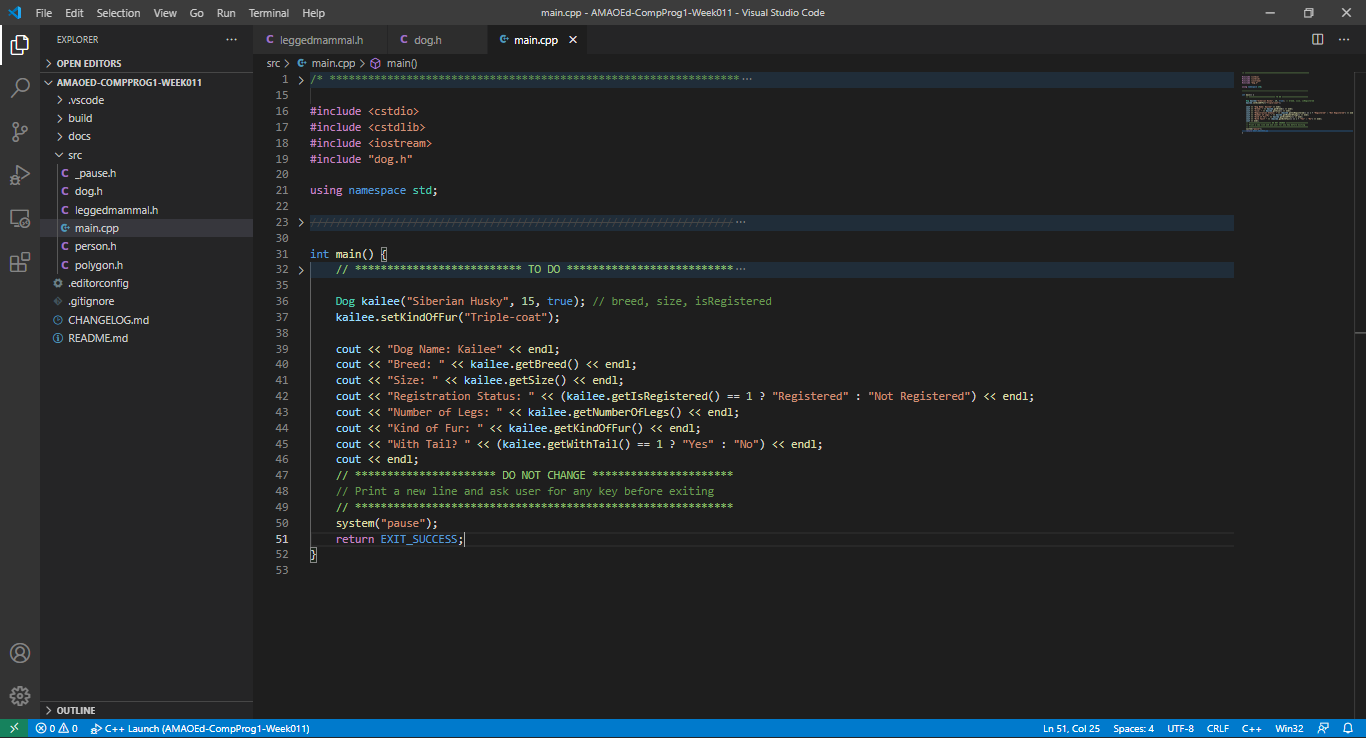
* **leggedmammal.h**



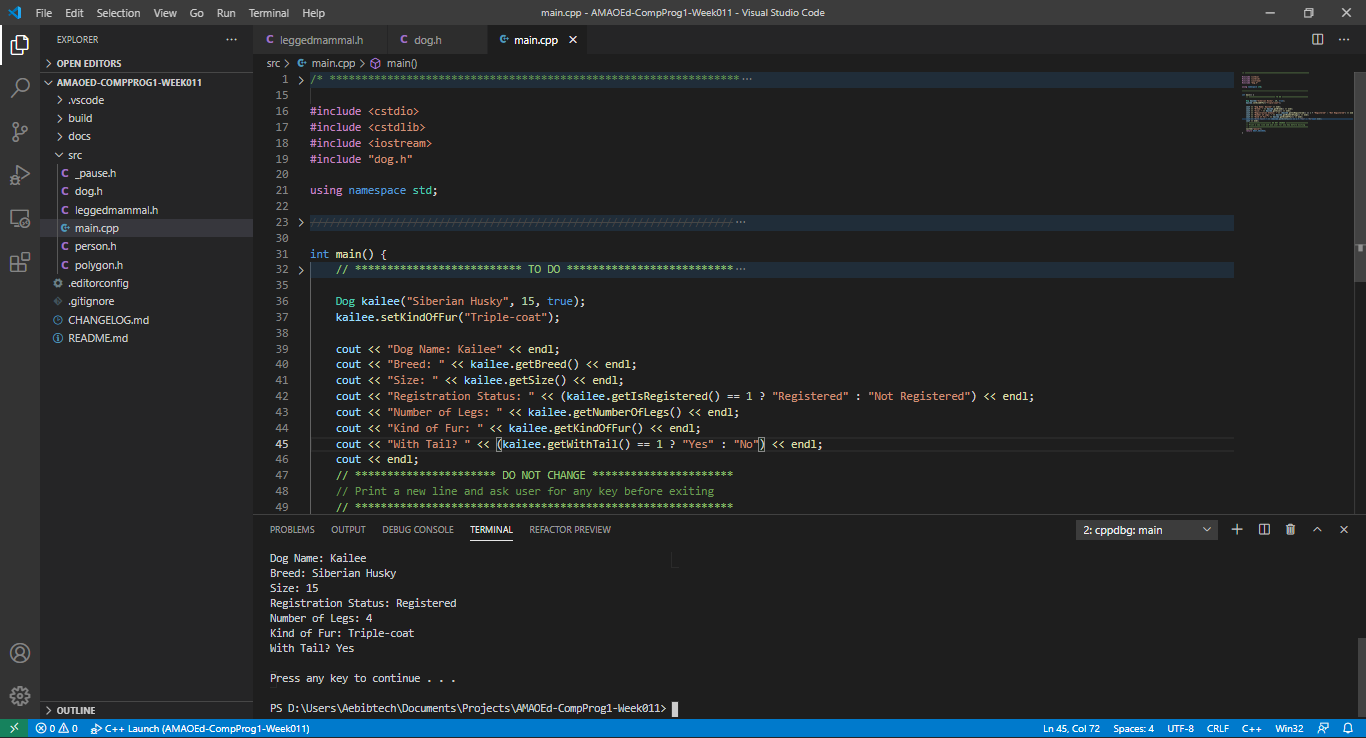
* **dog.h**



* **main.cpp**

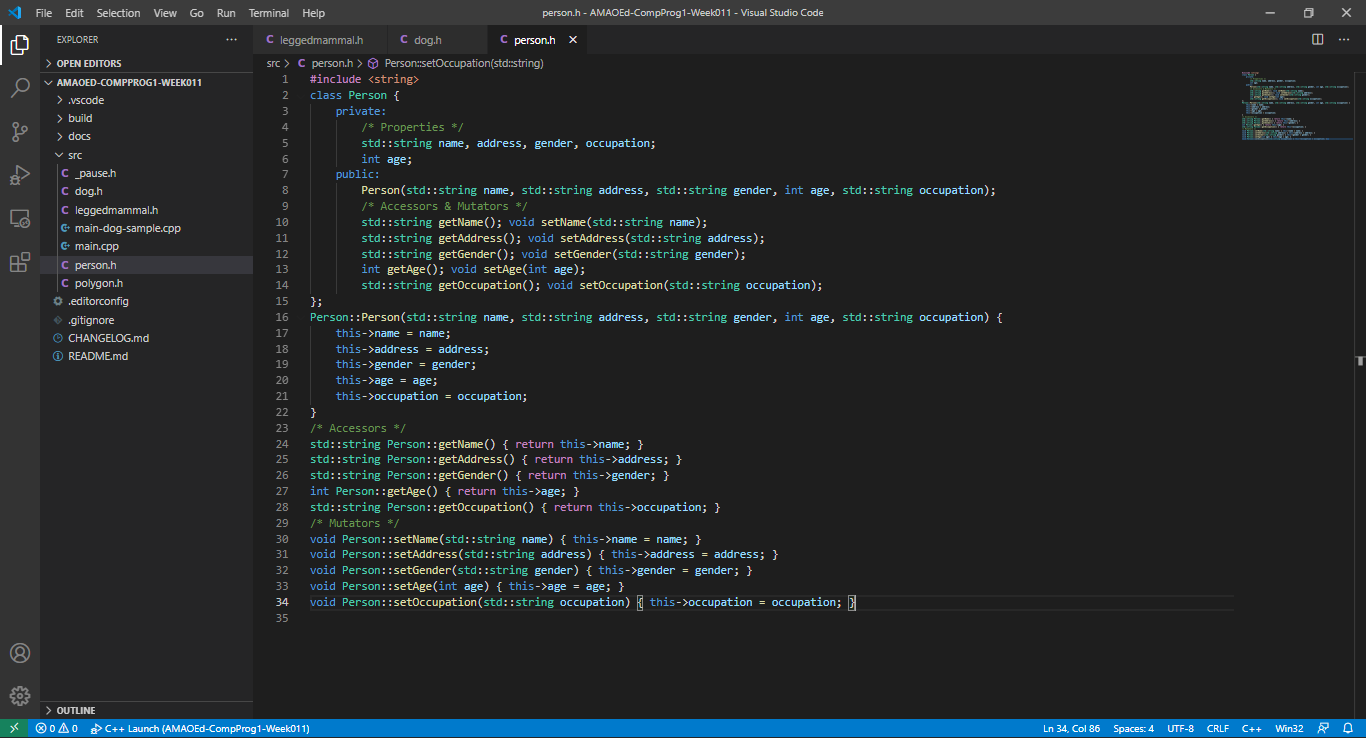


* **Output**

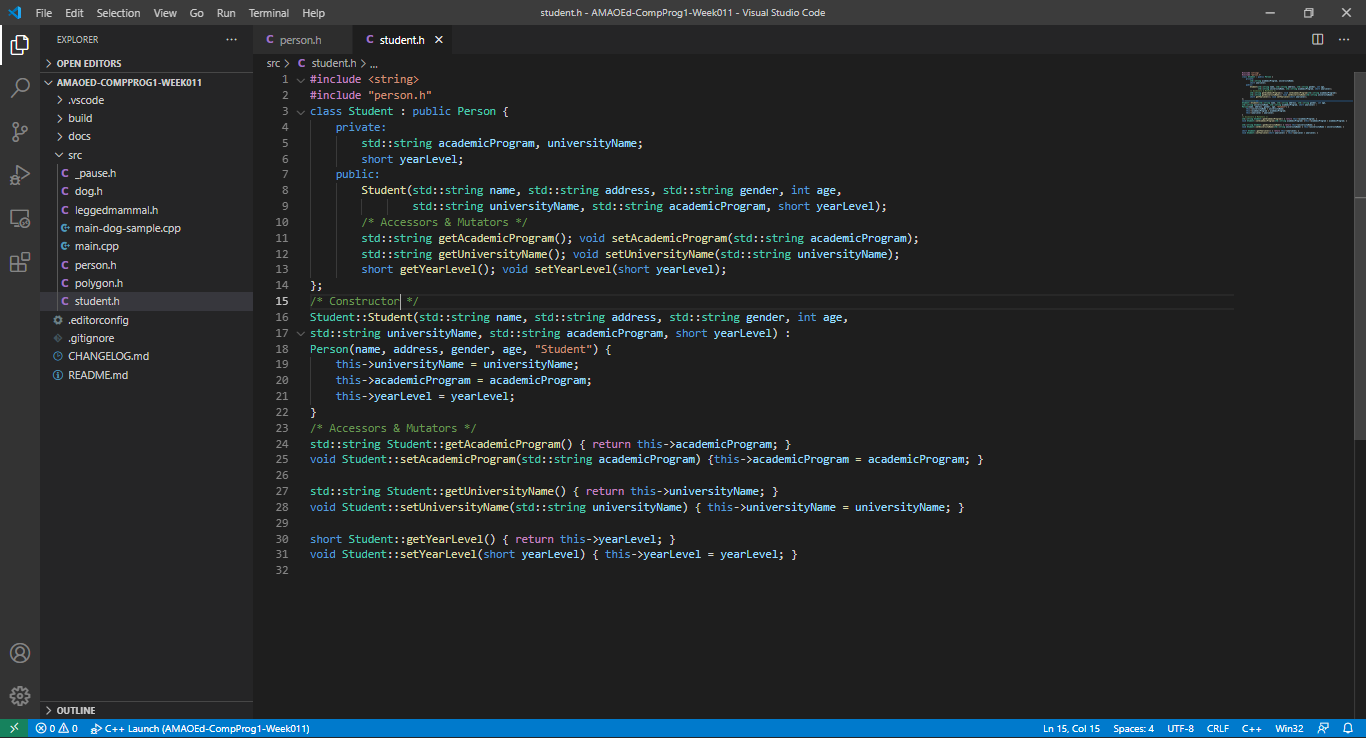


1. Write a class that extends the Person class from the previous laboratory exercise. The class will represent a Student. Consider the academic program, year in college and enrolled university. Initialize all the properties of the parent class in the new constructor. This time, promote the use of accessors and mutators for the new properties. Instantiate a Student object in the main function and be able to set the values of the properties of the Student object using the mutators. Display all the properties of the Student object using the accessors.

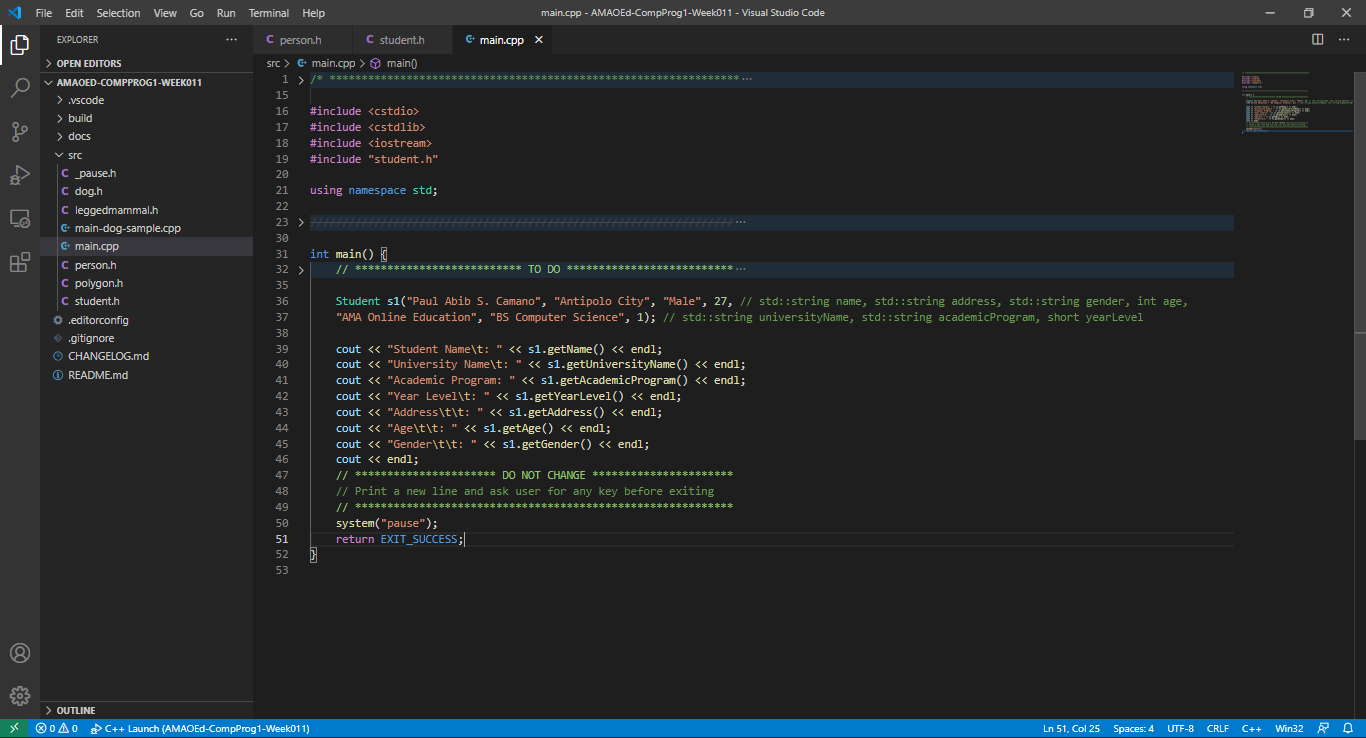
**person.h**



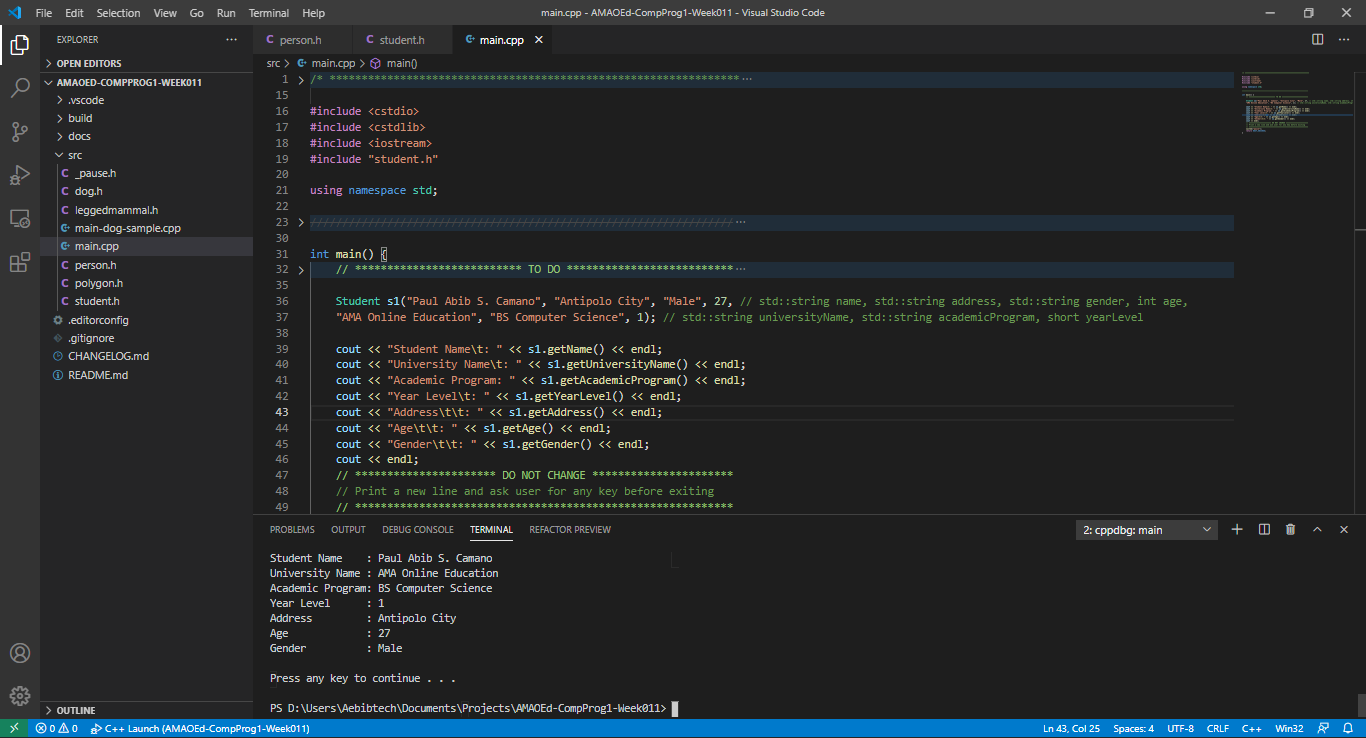
**student.h**



**main.cpp**



**Output**



1. What can you conclude from this activity?

I can conclude from this activity that the principles of object-oriented programming help programmers in grouping objects based on kind and functionality (inheritance), in hiding implementation details (encapsulation + abstraction) and in implementing a certain method with different parameters (polymorphism).