Encapsulation in coding is the process of enclosing data within the class that needs access to it. Variables and methods are made private to prevent other parts of the program from accessing the data. When other parts of the program need access to the data, they can be accessed through the class using public methods. Encapsulation has several benefits. It prevents data from being modified by other parts of the code. It assists in making the code modular so that if changes need to be made, they can be made within the classes without affecting the rest of the program. In the scripture memorization program that I completed this week, I used encapsulation to keep the reference and text of a scripture private, while using a public method to get the data and display it to the rest of the program. In the following code, encapsulation was used to keep the references private, then the necessary information was shared with the rest of the program using the public method.

private string \_book;

private int \_chapter;

private int \_verse;

private int \_startVerse;

private int \_endVerse;

public string GetDisplayText()

{

if (\_endVerse == 0)

{

return $"{\_book} {\_chapter}:{\_verse}";

}

else

{

return $"{\_book} {\_chapter}:{\_startVerse}-{\_endVerse}";

}

}