Week 03-CSE 210: Programming with Classes

What is encapsulation and why is it important?

Abstraction is separating something we want to do into segments, being able to visualize or examine what we want to do according to its properties and reasons. Applying abstraction helps us to be able to highlight the most appropriate things for what we are wanting to do, separating that activity or task into parts. For example, this week's daily program, dividing the different tasks to be carried out and saved in the program into segments, helps to not only be able to do it more easily but also to be able to change a function without ceasing to carry out the main thing. This can mainly be applied in large work scales such as studios. Encapsulation is to separate a function or program so that it works carefully in a way, which helps us to prevent other functions of the program from interrupting it, providing communication with other parts of this program. As a benefit it helps us to be able to see clearly and protect with limits the access to those variables and methods of the classes. As an example we can see this: using System;

public class Scripture

{

private List<Word> \_scripture = new List<Word>();

private int \_scriptureWordCount;

Random random = new Random();

public void \_storeScripture(string scpt)

{

string[] words = scpt.Split(" ");

\_scriptureWordCount = words.Length;

foreach (string word in words)

{

Word newword = new Word();

newword.\_set(word);

\_scripture.Add(newword);

}

}

that the encapsulation of private and public of some of these classes is used so as not to interfere in others, setting a limit.