1. Stack is used for static memory allocation and manage sequences as first in last out. Heap for dynamic memory allocation without any order/sequence. The reference type allocated in heap memory but the reference value portion will be located on stack. And if we assign to one another (eg student s1 = s).. we passing the same memory location in stack which is the same memory location in heap. If the memory is not needed.. Garbage collector will delete it.
2. Variables that store data are called value types. Value types are stored on stack. They contain the actual values.

Example: int, enum, structs.

Variables that store reference to actual data are called Reference types. Reference types stored on heap but contain the address on heap.  
Example: class, interface, delegate, string, object, Array

1. In the first one we got 3 because x and y are value types so x holds its own value which is found in stack. Value types are not depend on another.

For the second one, because x and y points the same value which is in the Heap. When we apply the assignment operator to reference types we are redirecting what the variable points to in memory (in heap). The value which is referred (MyValue) can be changed but x, y are pointing to same reference variable value.