1. ***Descrivere le differenze tra Reference Type e Value Type***

*Sono dei valori che vengono associati alla variabile e si differenziano sostanzialmente per la parte di memoria occupata. I value type assumono direttamente il valore che gli viene assegnato che sarà nella memoria chiamata Stack; esempi di variabile sono int,double,bool,bit.. Inoltre non possono essere null, devo inizializzarli subito.*

*I reference type invece non hanno un valore direttamente nello stack ma puntano a un valore nello Heap. Possono dunque inizialmente essere dichiarati ma non assegnati,avranno valore null. Esempi di ref type sono le String,che però hanno caratteristiche anche dei value type.*

***2. Cosa è un delegate?***

Un delegate è un metodo che viene delegato da un altro metodo, riprende la funzione dei puntatori di C#

Delegate void Mydelegate(int i);

Creandolo,il compilatore crea in automatico una classe System.Delegate. Io quindi andrò a istanziarlo come fosse una classe

MyDelegate d = new MyDelegate(MetodoX);

Di fatto,ha come parametro un altro metodo di cui fa il delegate.

Void MetodoX(int i);

Richiamo il delegate che esegue il metodo : d(2);

***3. Definire come costruire il design pattern Factory*  e qual è la sua utilità**

I design pattern sono una soluzione a un problema ricorrente. Il Factory fa parte dei Pattern Creazionali e serve per creare delle entità in base a un parametro di distinzione: è una classe di controllo! Questo è utile perché il main non sa quali entità vi sono nel programma,quindi serve anche per nasconderne alcune parti all’esterno. Io infatti avrò un interfaccia che verrà implementata dalle altre entità,che verrà richiamata del factory e mi permetterà di creare diverse entità in base al parametro