Strings

* String is a class.
* String is a Reference type.
* Instances of strings are objects.
* Is an array of characters.
* Always loaded in special location in a heap called as Intern pool.
* Every string in intern pool will be unique (i.e.) linked not created if same object.
* String objects stored in intern pool cannot be changed, they are immutable.
* Strings are treated as value type.
* System.Int32, string are alias to String.
* Integers are value types so can’t be set to null whereas Strings can be set to null since they are objects (i.e.) their references can be set to null.

StringBuilder

* Doesn’t go to intern pool straightly to heap.
* String.Format(“{0}, {1}, {3}”, “A”, “B”, “C”) same as $”ABC”
* String Builder Measured in No of characters and not on bits and bytes and its not fixed value its dynamic.

Exceptions

* At runtime something goes wrong
* Application based Exceptions (business logic)
* System Exceptions (Hardware/OS based)
* System.Exception is the base file for all exceptions.
* Has inbuilt Exception Manager tool which monitors stack for exceptions.
* Try is used for monitoring code under execution.
* Catch block is called only when there is an error.
* Finally is a block that is normally used for cleaning the very costly blocking resources.
* A single try can have multiple catch and one finally.
* Catch block is optional.
* A try should have either catch or finally.
* Every try need not have catch.
* One try can have multiple catch.

Members/Properties of an Exception

* Message
  + read only property, it gives error description.
* StackTrace
  + Is a list of methods through which stack is passed.
* TargetSight
  + The TargetSight is the actual method where the exception is raised.
* InnerException
  + An exception object inside another exception object is known as InnerException.
* Source
  + Returns the name of the assembly where the exception is raised.

(Exceptions doesn’t have methods)

try

{

//code here – business logic

}catch(Exception err)

See pic

A try can have multiple catch

Catches in reverse order of inheritance B A exception

VM

* Security Provider
* Exception Manager

Question paper Important

* Bubble sort
* Strings and substrings
* Overriding
* Interface
* For loop from entry test not pyramid :P not pattern
* Abstract classes
* What is polymorphism?
* StringBuilder (doubt)
* Exception Handling
* IF exception is not caught in inner try block then the exception will be caught in the outer try block (Throw err watch out bounce)