**S18: String and StringBuilder**

**String(class) Or string(datatype): -** its collection of character and its immutable (Unchange). **Its reference** type which store the value in **heap memory**. Always it create new memory block in heap memory. Use more memory. 2 times modification then used. To string method is used to get string from string builder.

Eg:- string name = “Imran”.

name = name + “Shaikh”.

name = name + “Pune”.

Imran |1001

Imran Shaikh |1002

Imran Shaikh Pune|1003

Heap Memory

**String Builder:-**  its mutable(change). Its declare like class. Stringbuilder is used to create string builder. Manipulation of string is easy. Used less memory. It present in System.Text class. Its dynamic object. Dosnt create new object previous object is used. More than 2 modification used this. By default its 16 character. And also you can give the capacity.

Eg:- StringBuilder stringBuilder = new StringBuilder(“Imran”, 50>> Size of SB);

stringBuilder.Append(“Shaikh”);

stringBuilder.Append(“Pune”);

Imran Shaikh Pune | 1001

Heap Memory

* **Methods: Append ( ):-** used to add string in string builder in front of existing string.
* **AppendLine( ):-** Add string in new Line.
* **Append Format:-** add string value with the specific format.
* **Insert:-** when you want to insert new string at some position by using index value then used this insert.
* **Remove:-** remove the specific string from the stringbuilder.

For checking time of execution we used **stopwatch** and its present in System.diagnostic package or module. Start method is used to start the timer and stop is used to stop the timer.

S18\_\_StringAndStrinBuilder.cs

using System;

using System.Collections.Generic;

using System.Text;

using System.Diagnostics; //checking execution time

namespace OOPS\_\_AllSession

{

class S18\_\_StringAndStrinBuilder

{

public void StringManupulation()

{

string names = "Abhilasha";

Stopwatch timer = new Stopwatch();

timer.Start();

for (int i = 0; i < 10; i++)

{

names = names + names;

Console.WriteLine("Names Are: " + names);

}

timer.Stop();

Console.Write("\nTime For String is : " + timer.ElapsedMilliseconds + "\n\n");

}

public void StringBuilder()

{

StringBuilder stringBuilder = new StringBuilder("Amit Kumar");

Stopwatch timer = new Stopwatch();

timer.Start();

for (int i = 0; i < 10; i++)

{

stringBuilder.Append("Deshmukh");

Console.WriteLine("\tNames Are: " + stringBuilder);

}

timer.Stop();

Console.Write("\nTime For StringBuilder is : " + timer.ElapsedMilliseconds);

}

public void StringBuilder\_Methods()

{

StringBuilder stringBuilder = new StringBuilder("\n\nAmit Kumar Deshmukh");

stringBuilder.Insert(4, "ee");

stringBuilder.AppendLine("Patil");

stringBuilder.Remove(6, 3);

stringBuilder.AppendFormat("{0:X}", 25);

Console.WriteLine("\tNames Are: " + stringBuilder);

}

}

}

OopsSessions.cs

using OOPS\_\_AllSession;

using System;

using static OOPS\_\_AllSession.S11\_\_ClassAndTypes;

namespace Oops\_\_AllSession

{

class OopsSessions

{

static void Main(string[] args)

{

Console.WriteLine("\*\*\*\*\*\*\*\*Welcome To Main Method\*\*\*\*\*\*\*\*\*\*\*");

S18\_\_StringAndStrinBuilder stringBuilder = new S18\_\_StringAndStrinBuilder();

stringBuilder.StringManupulation();

stringBuilder.StringBuilder();

stringBuilder.StringBuilder\_Methods();

}

}

}