**StringBuffer and StringBuilder** in java–

**StringBuffer** – java string is immutable (we already know that) a string buffer is similar to string but with more functionality. A string buffer is mutable. And its length and sequence content can be modified at any time by using specific method calls.

Multiple threads can use string buffer safely. The methods are synchronized as required such that all operations on any specific instance behave as if they occurred in any sequential order consistent with the order of the method calls made by any of the individual threads involved.

**StringBuffer class constructors:**

* StringBuffer() – creates an empty string buffer with an initial capacity of 16 characters.
  + StringBuffer sb = new StringBuffer();
* StringBuffer(int capacity) - creates an empty string buffer with specified initial capacity.
  + StringBuffer sb = new StringBuffer(25);
* StringBuffer(CharSequence seq) – creates string buffer with the same characters as the specified CharSequence.
  + CharSequence seq = “abhisek kumar”;
  + StringBuffer sb = new StringBuffer(seq);
* StringBuffer(String str) – creates a string buffer that is initialized by the contents of the specified string.
  + String str = “abhisek kumar”;
  + StringBuffer sb = new StringBuffer(str);

When an operation involving a source sequence occurs (for ex – inserting or appending from a source sequence), the string buffer does nor synchronize on the source. It only synchronizes on the strung buffer executing the operation.

**StringBuilder** – an equivalent class, built for use by a single thread, has been instructed to JDK StringBuffer class. The stringbuilder class is consistent with stringbuffer, but not synchronization is guaranteed. This class intended to be used as a drop-in substitute for strringbuffer in places where a single thread previously used the stringbuffer.

**Stringbuffer class constructors:**

* StringBuilder() – creates an empty string builder with an initial capacity of 16 characters.
  + StringBuilder sb = new StringBuilder();
* StringBuffer(int capacity) - creates an empty string buffer with specified initial capacity.
  + StringBuffer sb = new StringBuffer(25);
* StringBuilder(CharSequence seq) – creates string buffer with the same characters as the specified CharSequence.
  + CharSequence seq = “abhisek kumar”;
  + StringBuilder sb = new StringBuilder(seq);
* StringBuuilder(String str) – creates a string buffer that is initialized by the contents of the specified string.
  + String str = “abhisek kumar”;
  + StringBuilder sb = new StringBuilder(str);

Every stringbuffer and stringbuilder has a capacity. It is not necessary to assign a new internal buffer array as long as the length of the character sequence in the stringbuffer, or stringbuilder does not surpass the capacity. When the internal buffer overflows, it is automatically increased in size.

It is recommended that the string builder class be used instead of string buffer whenever possible because it is faster in most implementations. String Builder instances, on the other hand, are not secure to use by multiple threads. If such synchronization is needed, string buffer is the best choice.