In Java, strings are immutable, meaning their value cannot be changed once created. Modifying a string creates a new string in memory. For mutable strings, use StringBuilder (faster but not thread-safe) or StringBuffer (thread-safe but slower).

String s1 = "Scaler"; // Using string literal

String s2 = new String("InterviewBit"); // Using new keyword

\*\*\* Scanner sc = new Scanner(System.in);

sc.next(): Reads a single word from the input (delimited by spaces).

sc.nextLine(): Reads an entire line of text, including spaces, until the end of the line.

**Java StringBuffer Class**

StringBuffer s = new StringBuffer("Hello");

System.out.println(s); //Hello

System.out.println(s.toString()); //Hello, as s itself is not a string, it's a sequence of characters

s.append(" World");

System.out.println(s);

s.insert(6, "hi"); //insert hi from index 6, till index 4 Hello is present now index 5 is space then from index 6 insert hi

System.out.println(s); //Hello whiorld

s.replace(0, 1, "A");

System.out.println(s); //Aello whiorld

s.delete(0, 2); //o included, 2 excluded

System.out.println(s); //llo hiWorld

s.reverse();

System.out.println(s.toString()); //llo hiWorld

**Java StringBuilder Class**

StringBuilder s = new StringBuilder ("Hello");

System.out.println(s); //Hello

System.out.println(s.toString()); //Hello, as s itself is not a string, it's a sequence of characters

s.append(" World");

System.out.println(s);

s.insert(6, "hi"); //insert hi from index 6, till index 4 Hello is present now index 5 is space then from index 6 insert hi

System.out.println(s); //Hello whiorld

s.replace(0, 1, "A");

System.out.println(s); //Aello whiorld

s.delete(0, 2); //o included, 2 excluded

System.out.println(s); //llo hiWorld

s.reverse();

System.out.println(s.toString()); //llo hiWorld