

✓ equals();
↳ method

✓ == → operator

to compare int → 5 == 5
a == b

✓ equals() → compares the content/value of string

✓ == → compares the address of the two strings → when comparing objects

String str1 = "xyz"
String str2 = "xyz"

str1 == str2 → True

String str3 = new String("xyz");
String str4 = new String("xyz");

str3 == str4 → false

str1 == str3 → false

CHARAT → num[i]

String name = "abc";

name.charAt(0) → // a

SUBSTRING

↳ subString(startIndex) → included

→ substring(startIndex, endIndex); → excluded

String courseName = "Data Structures";

String st = courseName.substring(0, 4) → Data

(0, 5) → // "Data_"

String st = courseName.substring(5); → // Structures

(4); → // _structures

SPLIT

```
String courseName = "Data Structures And Algorithms";
```

```
String[] strArr = courseName.split(" ");
```

↓
strArr[0] ⇒ "Data"

strArr[1] = "Structures"

String courseName = "Data⁰xyzStructures¹xyzAnd²"

```
courseName.split("xyz");
```

TRIM

```
String name = " --- xyz --- ";
```

```
name.trim(); // "xyz"
```

StringBuffer & StringBuilder → mutable variation of String

"Name"

String output = "N" + "m" (or) output.concat("m");

StringBuffer → Thread-safe → Slower ✓
✓StringBuilder → Not thread-safe → faster } Strings are much slower

StringBuilder strBui = new StringBuilder("xyz");

StringBuffer strBuf = new StringBuffer("xyz");

APPEND →

strBui.append("abc");

INSERT

StringBuilder strBui = new StringBuilder("Welcome");

strBui.insert(3, "xyz"); // "Welxyzcome"

REVERSE

strBui.reverse()

ToString

StringBuilder.toString(); → String

DELETE → .delete(startIndex, endIndex)

str = Welcome → str.delete(1, 3) → Wcome

DELETE CHARAT

str.deleteCharAt(1) → // Welcome

PROBLEM 1:

String str = "abccba" → {abc, cba} → equal
→ "abcba"

→ abcdef → {abc, fed} → not equal

Given a string, find if it is a palindrome

start ↓ end ↓
"abccba"

while (start < end) {

start == end ✓

start++;

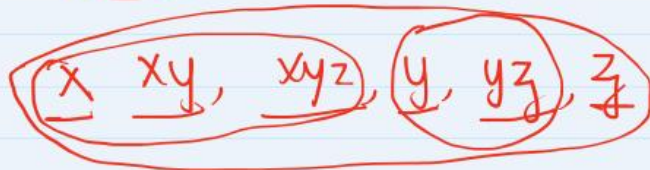
end--;

}

Problem 2: Print all the substrings of a string

eg: str = "xyz"

0 1 2 \rightarrow x, y, z, xy, yz, xyz



Problem 3: Is Anagram

str1 = "xyz" str2 = "yzx" \rightarrow sort

yxixiz ✓

yxixiz

str1 = "aabba"

str2 = "acbbb"

3a 2b X

2a 3b

① sort & compare

str1 - xyz

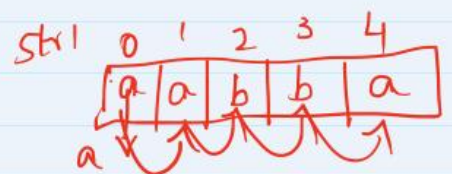
str2 \rightarrow xyz

} compare both

eg 2: str1 \rightarrow aabba

str2 \rightarrow acbbb

} \rightarrow compare



str2 \rightarrow ~~acbbb~~ \rightarrow (-1)

\rightarrow indexOf(a) \rightarrow 0

②

0	1
2	
a	b

[3
a

] → space str 1 } → compare
] → str 2

③ Str 1 Str 2
 ↓ to CharArray
 ✓ Char[] CharArr → situate

