

07

MARCH
THURSDAY
DAYS 067-299

| S | M | T | W | T | F | S |
|----|----|----|----|----|----|----|
| 31 | | | | 1 | 2 | |
| 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| 17 | 18 | 19 | 20 | 21 | 22 | 23 |
| 24 | 25 | 26 | 27 | 28 | 29 | 30 |

MARCH '24

String → ^① Objects are used to store ^① strings
of text.

A String Variable contains a collection of characters surrounded by double quotes?

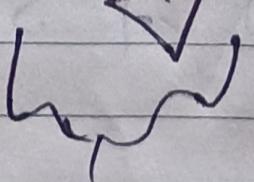
String name;

name = new String ("Shubham")

* String is a class but can be used like a data type

String is Class

String name = "Harry";



object.

* String are Immutable.

| S | M | T | W | T | F | S |
|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | |
| 7 | 8 | 9 | 10 | 11 | 12 | 13 |
| 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| 21 | 22 | 23 | 24 | 25 | 26 | 27 |
| 28 | 29 | 30 | | | | |

MARCH
FRIDAY

068-298 DAYS

08

Different ways to print in Java.

System.out.print() → No New Line

System.out.println() → New line added.

System.out.println()

System.out.format()

System.out.print("%.", ch)

%d for int } Format

%f for float } Specifier

%.C for char }

%.S for String }

String Methods }

String methods operate on Java strings. They can be used to find length of the string, convert to lowercase, etc.

Method & method is a block of code which only runs when it is called.

09

MARCH
SATURDAY
DAYS 069-297

| S | M | T | W | T | F | S |
|----|----|----|----|----|----|----|
| 31 | | 1 | 2 | | | |
| 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| 17 | 18 | 19 | 20 | 21 | 22 | 23 |
| 24 | 25 | 26 | 27 | 28 | 29 | 30 |

MARCH 24

Some of the commonly used String methods are

String name = "Harry"

Variable

- 1) \rightarrow name.length() → Returns length of String name. ✓
(5 in this case)
- 2) name.toLowerCase() - Returns a new string which has all the lowercase characters from the String name.
- 3) name.toUpperCase() Return a new string which has all the uppercase characters from the String name.
- 4) name.trim() Returns a new string after removing all the leading and trailing spaces from the original String.

10 SUNDAY

2024

| S | M | T | W | T | F | S |
|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 7 | 8 | 9 | 10 | 11 | 12 | 13 |
| 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| 21 | 22 | 23 | 24 | 25 | 26 | 27 |
| 28 | 29 | 30 | | | | |

APRIL 24

01 2 3 4 5 6 7 8
5 11 0 3 4 A Y I

MARCH
MONDAY
071-295 DAYS

11

2024

5) name.substring (int start)

Returns a Substring from start to the end substring (3) ↘ returns "ry"

{ Note that index starts from 0 }

6) name.substring (int start, int end)

Returns a Substring from start index to the end index ↘ Start index is included and end is excluded.

7) name.replace (char start, char end)

Returns a new string after replacing e with p. Happy is returned in this case.

{ Argument taken in string }
Inverted comma

replace method is a Case sensitive

If you want to replace one or more char then use (" ") double inverted commas

12

MARCH
TUESDAY
DAYS 072-294

| S | M | T | W | T | F | S |
|----|----|----|----|----|----|----|
| 31 | | | | 1 | 2 | |
| 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| 17 | 18 | 19 | 20 | 21 | 22 | 23 |
| 24 | 25 | 26 | 27 | 28 | 29 | 30 |

MARCH 24

replace method change or replace
when char is sequence way

String name = "Sheb Lam"

sout = (name.replace("S", "Su"),
 "ad")

then ~~#~~ no replacement takes place

Output is same

"Shebham"

sout = (name.replace("Sh", "ad"))

Output "adubham"

| S | M | T | W | T | F | S |
|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | |
| 7 | 8 | 9 | 10 | 11 | 12 | 13 |
| 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| 21 | 22 | 23 | 24 | 25 | 26 | 27 |
| 28 | 29 | 30 | | | | |

SHUBHAM
↓
Index 0 1 2 3 4 5.6
MARCH
WEDNESDAY
073-293 DAYS

13

8) name.startwith()

Returns true if name starts with string "Sh" true in this case!

9) name.endswith("am")

Returns true if name ends with string "am" true in this case.

10) name.charAt(2) "Return character at a given index position
"h" is this case!

11) name.indexOf('l')! Returns character at chosen index position

"l" is this case!

12) name.indexOf("S", 3) →

Returns the index of the given string starting from the index 3 (int). -1 is returned in this case.

14

MARCH
THURSDAY
DAYS 074-292

Shubham
⑥ 1 2 3 4 5 6

| S | M | T | W | T | F | S |
|----|----|----|----|----|----|----|
| 31 | | | | | 1 | 2 |
| 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| 17 | 18 | 19 | 20 | 21 | 22 | 23 |
| 24 | 25 | 26 | 27 | 28 | 29 | 30 |

MARCH 24

out is -1 because character B is not found in substring from index 5; the 'indexOf' method returns ' -1 ' to indicate that character is not present in specified search range.

13) name.lastIndexOf("A")

Returns the last index of the given string before index 2

14) name.lastIndexOf("A", 5)

Returns the last index of the given string before index 5

15) name.equalsIgnoreCase("Shubham")

Returns true if the given string is equal to "Shubham" false otherwise

2024 to "Shubham" false otherwise

case sensitive