ABESIT
CRC-Training

String

ABESIT
CRC-Training

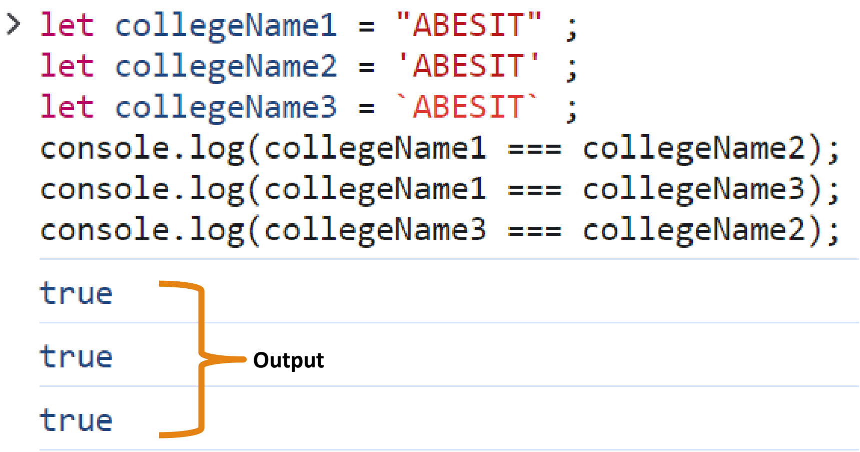
Strings in JavaScript

- In JavaScript, a string is a data type representing a sequence of characters that may consist of **letters, numbers, symbols, words, or sentences.**
- A JavaScript string is zero or more characters written inside quotes.
- **You can use single or double quotes:**
  - let collegeName1 = "ABESIT";   // Double quotes
  - let collegeName2 = 'ABESIT';   // Single quotes
  - let str = `ABESIT`   // Backticks

Create strings in JavaScript

- In JavaScript, we can create strings by wrapping the text inside
  - single quotes ('),
  - double quotes ("),
  - or backticks (`).
- Note - there is another way to create strings in JavaScript, which is via the **String() constructor**. It generate string as an object.

Example - String



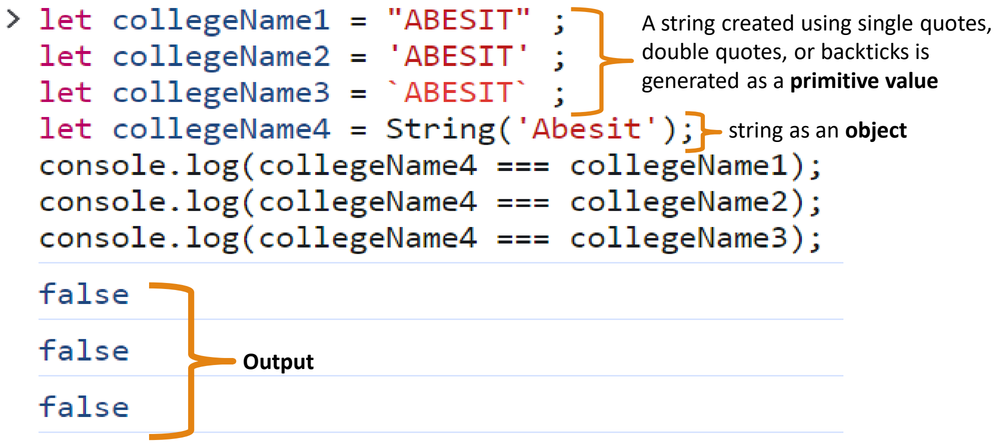
```
> let collegeName1 = "ABESIT" ;
  let collegeName2 = 'ABESIT' ;
  let collegeName3 = `ABESIT` ;
  console.log(collegeName1 === collegeName2);
  console.log(collegeName1 === collegeName3);
  console.log(collegeName3 === collegeName2);
```

```
true
true
true
```

Output

ABESIT
CRC-Training

Example - String



```
> let collegeName1 = "ABESIT" ;
  let collegeName2 = 'ABESIT' ;
  let collegeName3 = `ABESIT` ;
  let collegeName4 = String('Abesit');
  console.log(collegeName4 === collegeName1);
  console.log(collegeName4 === collegeName2);
  console.log(collegeName4 === collegeName3);
  false
  false
  false
```

A string created using single quotes, double quotes, or backticks is generated as a **primitive value**

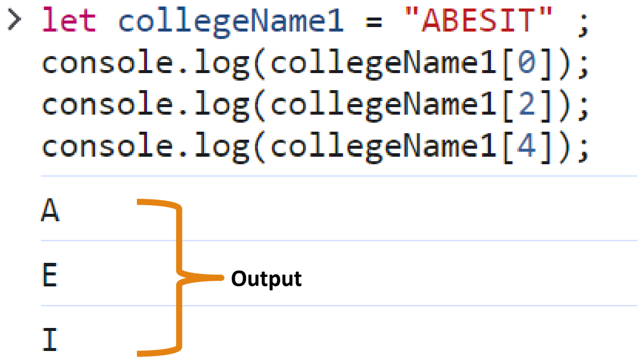
string as an **object**

Output

ABESIT
CRC-Training

Basic String Manipulation

String indexing - We can access each character inside a string through its numeric index – starting from zero – using bracket notation



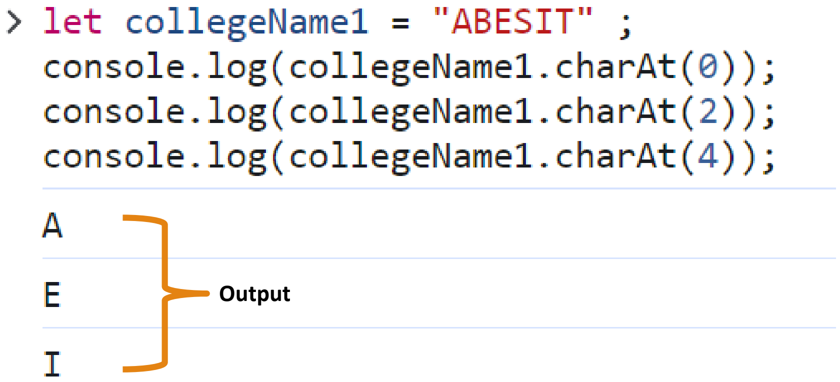
```
> let collegeName1 = "ABESIT" ;
  console.log(collegeName1[0]);
  console.log(collegeName1[2]);
  console.log(collegeName1[4]);
  A
  E
  I
```

Output

ABESIT
CRC-Training

Basic String Manipulation

We can use the **charAt() method** to get a specific character inside the string:



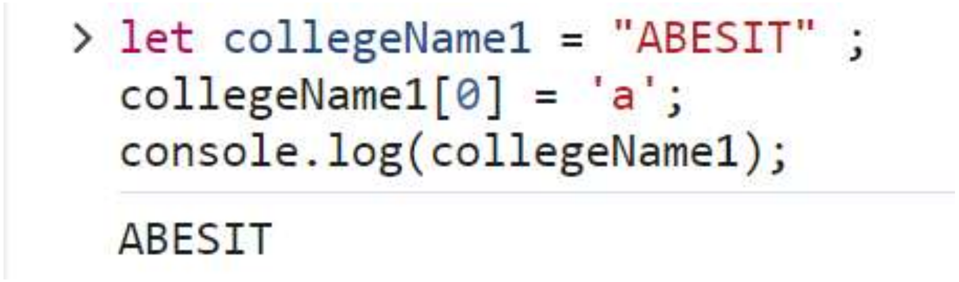
```
> let collegeName1 = "ABESIT" ;
  console.log(collegeName1.charAt(0));
  console.log(collegeName1.charAt(2));
  console.log(collegeName1.charAt(4));
  A
  E
  I
```

Output

ABESIT
CRC-Training

String is Imutable

- We **cannot mutate** a string



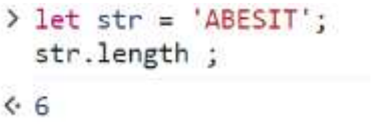
```
> let collegeName1 = "ABESIT" ;
  collegeName1[0] = 'a';
  console.log(collegeName1);
  ABESIT
```

ABESIT

CRC-Training

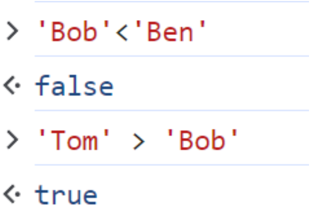
length property and String comparison

We get the number of characters contained in a string using the length property:



```
> let str = 'ABESIT';
  str.length ;
<· 6
```

String comparison – We can compare strings based on their **alphabetical order** and **length** using arithmetic **comparison operators**.



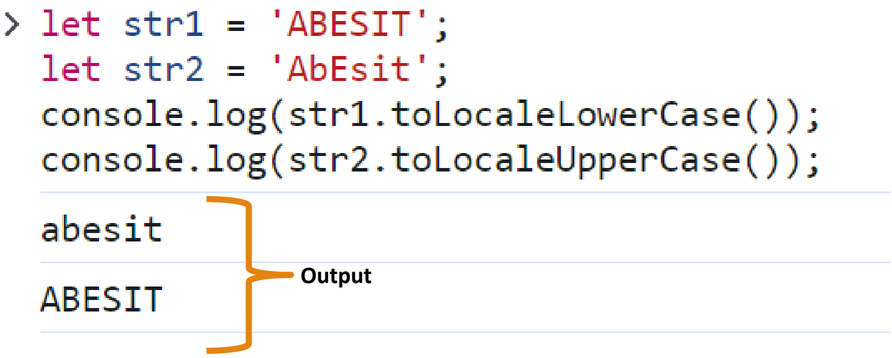
```
> 'Bob'<'Ben'
<· false
> 'Tom' > 'Bob'
<· true
```

ABESIT

CRC-Training

Common String Methods in JavaScript

toLowerCase() & toUpperCase() methods



```
> let str1 = 'ABESIT';
  let str2 = 'AbEsit';
  console.log(str1.toLocaleLowerCase());
  console.log(str2.toLocaleUpperCase());
  abesit
  ABESIT
```

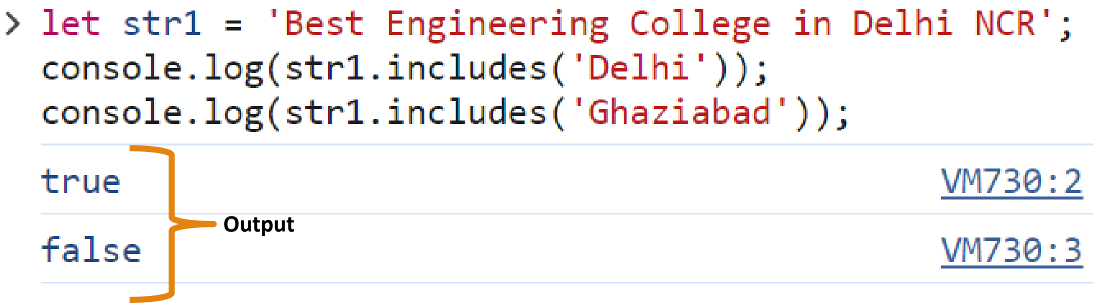
Output

ABESIT
CRC-Training

Common String Methods in JavaScript

includes() method - The includes() method checks if a specified string, passed as an argument, is present inside another string.

The search is case-sensitive and the return value is a boolean.



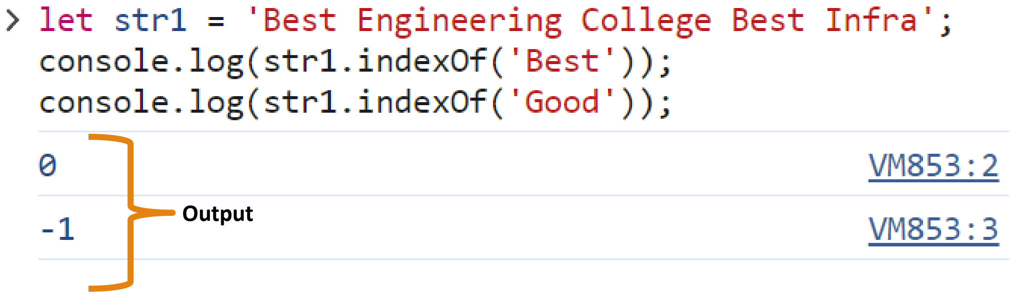
```
> let str1 = 'Best Engineering College in Delhi NCR';
  console.log(str1.includes('Delhi'));
  console.log(str1.includes('Ghaziabad'));
  true                                         VM730:2
  false                                        VM730:3
```

ABESIT
CRC-Training

Common String Methods in JavaScript

indexOf() methods - The indexOf() method searches for a substring and returns the first occurrence of the substring inside the calling string.

If the substring is not found, it returns -1.

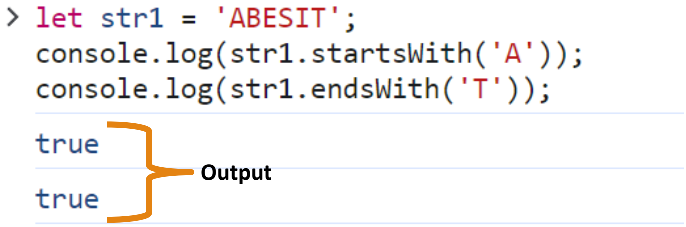


```
> let str1 = 'Best Engineering College Best Infra';
  console.log(str1.indexOf('Best'));
  console.log(str1.indexOf('Good'));
  0                                            VM853:2
  -1                                           VM853:3
```

Common String Methods in JavaScript

startsWith() & endsWith() methods

- The **startsWith()** method checks if a string begins with a specific sequence of characters and returns a boolean value.
- The **endsWith() method** checks if a string ends with a specific sequence of characters, returning a boolean value.

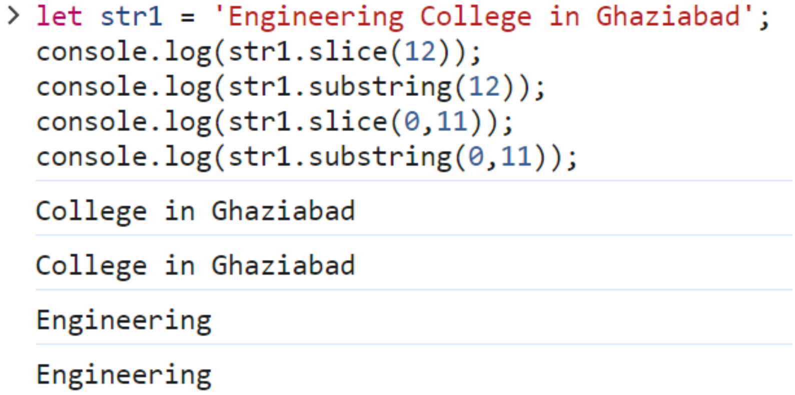


```
> let str1 = 'ABESIT';
  console.log(str1.startsWith('A'));
  console.log(str1.endsWith('T'));
  true
  true
```

Output

Common String Methods in JavaScript

slice() & substring() methods - The slice() and substring() methods pull a portion of a string, returning it as a new string.



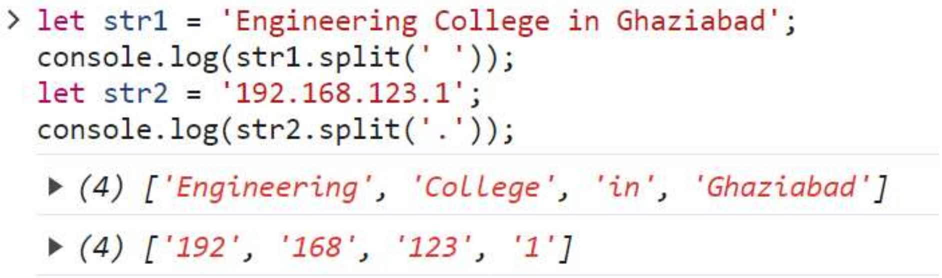
```
> let str1 = 'Engineering College in Ghaziabad';
  console.log(str1.slice(12));
  console.log(str1.substring(12));
  console.log(str1.slice(0,11));
  console.log(str1.substring(0,11));
  College in Ghaziabad
  College in Ghaziabad
  Engineering
  Engineering
```

Common String Methods in JavaScript

split() method –

- The split() method takes a **separator argument** and breaks a string up, according to the occurrence of the separator character inside the string.

- It returns an array of strings.



```
> let str1 = 'Engineering College in Ghaziabad';
  console.log(str1.split(' '));
  let str2 = '192.168.123.1';
  console.log(str2.split('.'));
```

```
▶ (4) ['Engineering', 'College', 'in', 'Ghaziabad']
▶ (4) ['192', '168', '123', '1']
```