



****

**INPUT/OUTPUT:**

**Version 3:**

\*\*\* start of Indexing Strings Demo \*\*\*

Type not empty string (q to quit):

lets start

Type the character position within the string:

1

The character found at position 1 is 'e'

Type not empty string (q to quit):

q

\*\*\* Start of Measuring Strings Demo \*\*\*

Type a string (q - to quit):

wer

The length of 'wer' is 3 characters

Type a string (q - to quit):

q

\*\*\* End of Measuring Strings Demo \*\*\*

\*\*\* Start of Copying strings Demo \*\*\*

Destination string is reset to empty

Type the source string (q to quit):

lets start

New destination string is 'lets start'

Destination string is reset to empty

Type the source string (q to quit):

this is new strng to test the code in the middle range

New destination string is 'this is new strng to test the code in the middle range'

Destination string is reset to empty

Type the source string (q to quit):

nice code lets try to break the code and see how many letters it can hold in one i guess this is enough to break it.

New destination string is 'nice code lets try to break the code and see how many letters it can hold in o'

Destination string is reset to empty

Type the source string (q to quit):

New destination string is 'e i guess this is enough to break it.'

Destination string is reset to empty

Type the source string (q to quit):

0123456789012345678901234567890123456789012345678901234567890123456789012345678901234567

New destination string is '012345678901234567890123456789012345678901234567890123456789012345678901234567'

Destination string is reset to empty

Type the source string (q to quit):

New destination string is '901234567'

Destination string is reset to empty

Type the source string (q to quit):

012345678901234567890123456789012345678901234567890123456789012345678901234567

New destination string is '012345678901234567890123456789012345678901234567890123456789012345678901234567'

Destination string is reset to empty

Type the source string (q to quit):

mix type testing 1234<>~?:L"

New destination string is 'mix type testing 1234<>~?:L"'

Destination string is reset to empty

Type the source string (q to quit):

New destination string is ' '

Destination string is reset to empty

Type the source string (q to quit):

New destination string is ''

Destination string is reset to empty

Type the source string (q to quit):

okay this works properly and code is quite good

New destination string is 'okay this works properly and code is quite good'

Destination string is reset to empty

Type the source string (q to quit):

q

\*\*\* End of Copying Strings Demo \*\*\*

**Version 2:**

\*\*\* start of Indexing Strings Demo \*\*\*

Type not empty string (q to quit):

q

\*\*\* Start of Measuring Strings Demo \*\*\*

Type a string (q - to quit):

0123456789

The length of '0123456789' is 10 characters

Type a string (q - to quit):

thisisnewtestcase

The length of 'thisisnewtestcase' is 17 characters

Type a string (q - to quit):

012345678901234567890123456789012345678901234567890123456789012345678901234567

The length of '012345678901234567890123456789012345678901234567890123456789012345678901234567' is 78 characters

Type a string (q - to quit):

'

The length of ''' is 1 characters

Type a string (q - to quit):

thisisnewusertryingtotestthecodewithvisualocdeeditor

The length of 'thisisnewusertryingtotestthecodewithvisualocdeeditor' is 52 characters

Type a string (q - to quit):

0123456789012345678901234567890123456789012345678901234567890123456789012345

The length of '0123456789012345678901234567890123456789012345678901234567890123456789012345' is 76 characters

Type a string (q - to quit):

6789001

The length of '6789001' is 7 characters

Type a string (q - to quit):

01234567890123456789012345678901234567890123456789012345678901234567890123456789001

The length of '012345678901234567890123456789012345678901234567890123456789012345678901234567' is 78 characters

Type a string (q - to quit):

The length of '9001' is 4 characters

Type a string (q - to quit):

-12

The length of '-12' is 3 characters

Type a string (q - to quit):

!thia is number two 2

The length of '!thia is number two 2' is 21 characters

Type a string (q - to quit):

The length of '' is 0 characters

Type a string (q - to quit):

q

\*\*\* End of Measuring Strings Demo \*\*\*

**Version 1:**

\*\*\* start of Indexing Strings Demo \*\*\*

Type not empty string (q to quit):

123456789

Type the character position within the string:

0

The character found at position 0 is '1'

Type not empty string (q to quit):

123456789

Type the character position within the string:

5

The character found at position 5 is '6'

Type not empty string (q to quit):

0123456789

Type the character position within the string:

9

The character found at position 9 is '9'

Type not empty string (q to quit):

abcdefghij

Type the character position within the string:

3

The character found at position 3 is 'd'

Type not empty string (q to quit):

012345678901234567890123456789

Type the character position within the string:

18

The character found at position 18 is '8'

Type not empty string (q to quit):

abcdefghijklmnopqrstuvwxyz

Type the character position within the string:

25

The character found at position 25 is 'z'

Type not empty string (q to quit):

34567

Type the character position within the string:

5

Too big... Position reduced to max. available

The character found at position 4 is '7'

Type not empty string (q to quit):

23456

Type the character position within the string:

-2

Too big... Position reduced to max. available

The character found at position 4 is '6'

Type not empty string (q to quit):

theGAME

Type the character position within the string:

k

The character found at position 0 is 't'

\*\*\* End of Fundamentals Strings to int Demo \*\*\*