

# Strings

String is data type that stores a sequence of characters.

## *Basic Operations*

- **concatenation**

"hello" + "world"     $\longrightarrow$     "helloworld"

- **length of str**

len(str)

# Indexing

**A p n a \_ C o l l e g e**  
0 1 2 3 4 5 6 7 8 9 10 11

```
str = "Apna_College"
```

```
str[0] is 'A', str[1] is 'p' ...
```

```
str[0] = 'B' #not allowed
```

# Slicing

## Accessing parts of a string

`str[ starting_idx : ending_idx ]` #ending idx is not included

`str = "ApnaCollege"`

`str[ 1 : 4 ]` is "pna"

`str[ : 4 ]` is same as `str[ 0 : 4 ]`

`str[ 1 : ]` is same as `str[ 1 : len(str) ]`

# Slicing

*Negative Index*

**A p p l e**  
-5 -4 -3 -2 -1

str = "Apple"

str[ -3 : -1 ] is "pl"

# String Functions

```
str = "I am a coder."
```

```
str.endsWith("er.") #returns true if string ends with substr
```

```
str.capitalize() #capitalizes 1st char
```

```
str.replace(old, new) #replaces all occurrences of old with new
```

```
str.find(word) #returns 1st index of 1st occurrence
```

```
str.count("am") #counts the occurrence of substr in string
```

# Let's Practice

**WAP to input user's first name & print its length.**

**WAP to find the occurrence of '\$' in a String.**

# Conditional Statements

## if-elif-else (SYNTAX)

if(condition) :

Statement1

elif(condition):

Statement2

else:

StatementN

# Conditional Statements

Grade students based on marks

`marks >= 90, grade = "A"`

`90 > marks >= 80, grade = "B"`

`80 > marks >= 70, grade = "C"`

`70 > marks, grade = "D"`



# Let's Practice

**WAP to check if a number entered by the user is odd or even.**

**WAP to find the greatest of 3 numbers entered by the user.**

**WAP to check if a number is a multiple of 7 or not.**