

PYTHON

center(), count()
encode(), endswith()

- Python String Methods
 - `str.center()`
 - `str.count()`
 - `str.encode()`
 - `str.endswith()`

- Python String Methods

- `str.center(width, fillchar)`

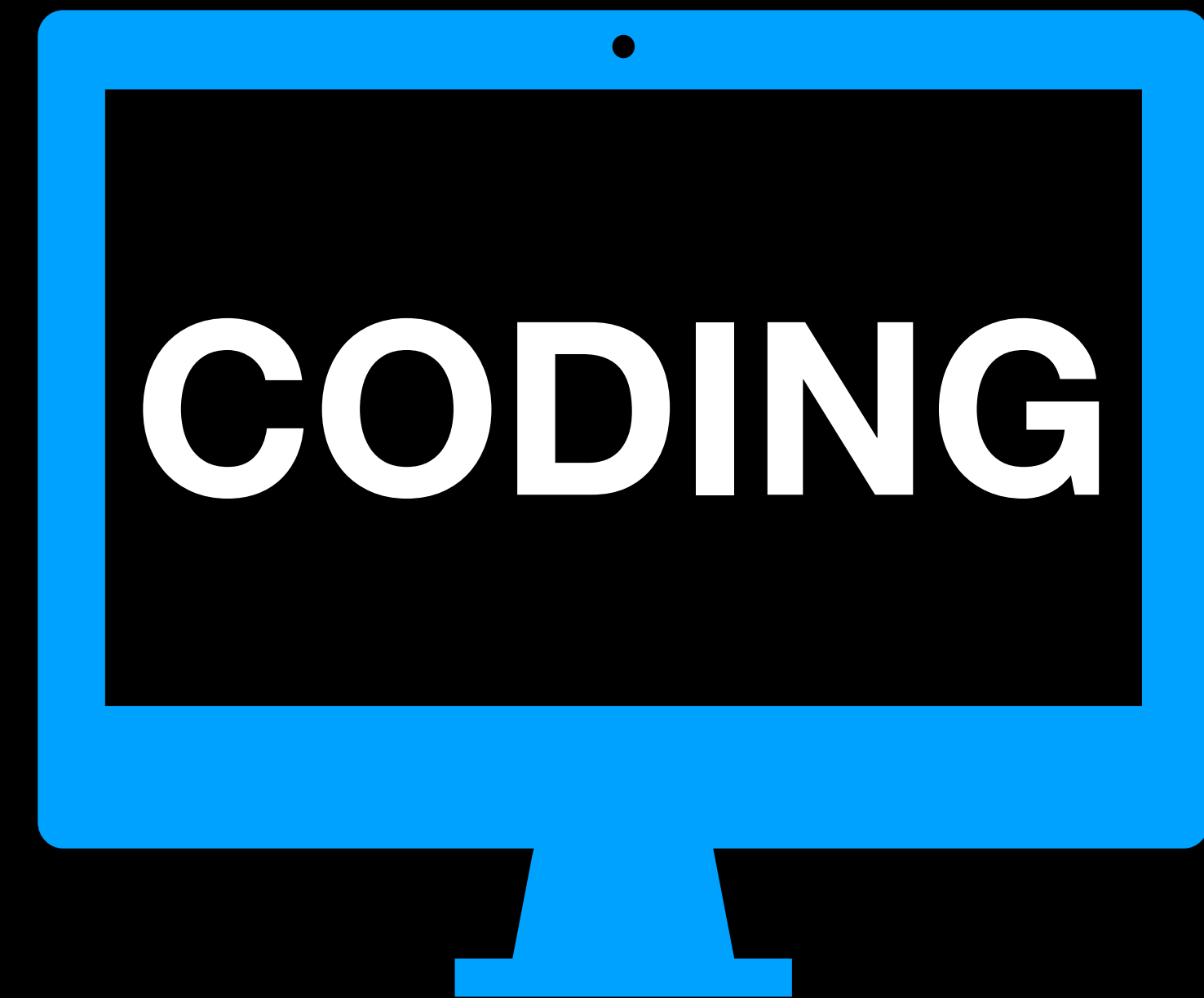
Where,

- ▲ `width(Required)` : Length of String after padding

- ▲ `fillchar(Optional)` : Character which need to be padded, Default padding value is space

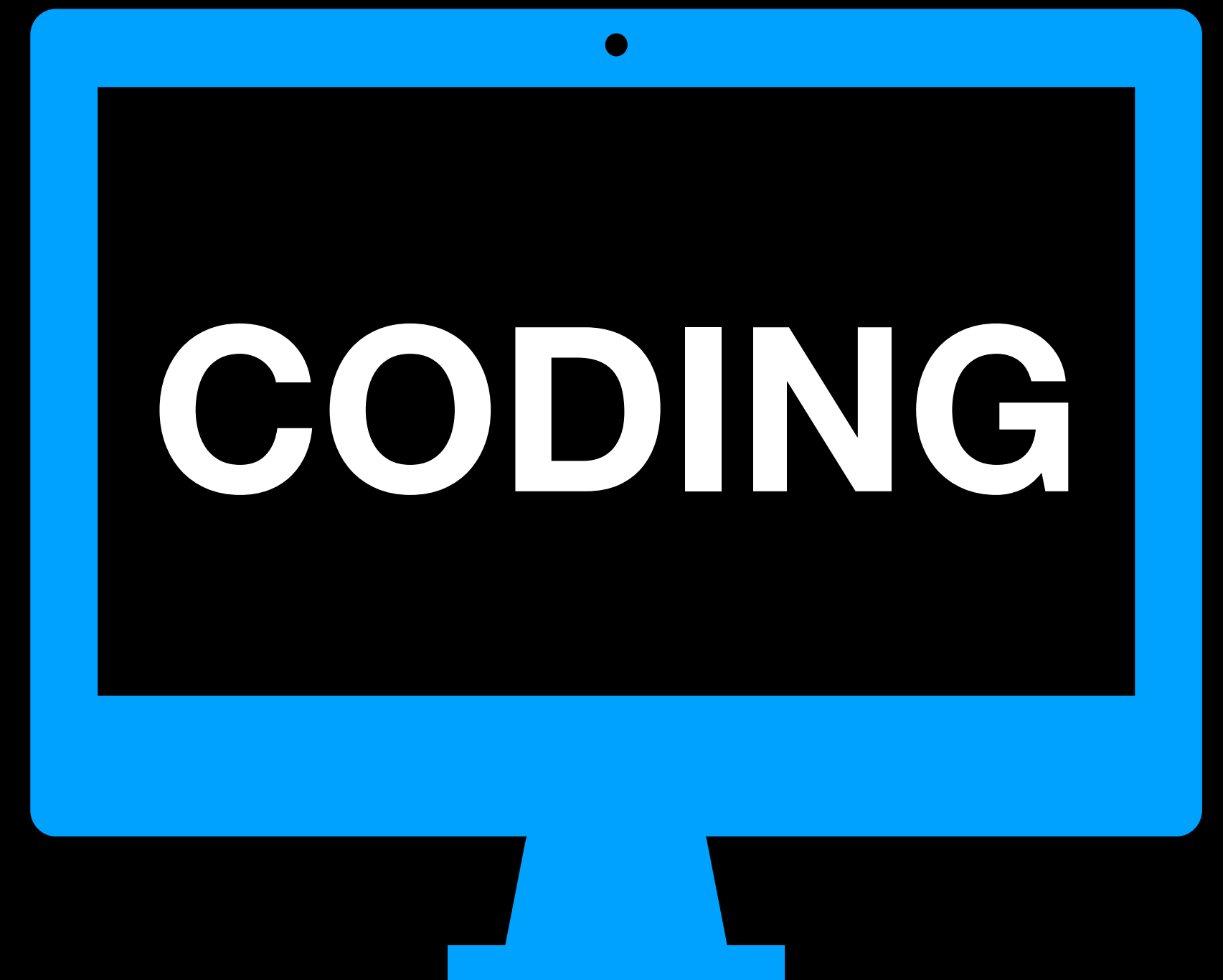
- Output : It returns the new string aligned by filling paddings left and right String

- Python String Methods
 - `str.center(width, fillchar)`



- Python String Methods
 - `str.count(value, start, end)`
 - ▲ `value(Required)` : String whose count is to be found
 - ▲ `start(Optional)` : Index from where search to be start
 - ▲ `end(Optional)` : Index from where search to be end
 - Output : It returns the total count of given sub-string

- Python String Methods
 - `str.count(value, start, end)`



- Python String Methods - TODO

- `str.encode(encoding, error)`

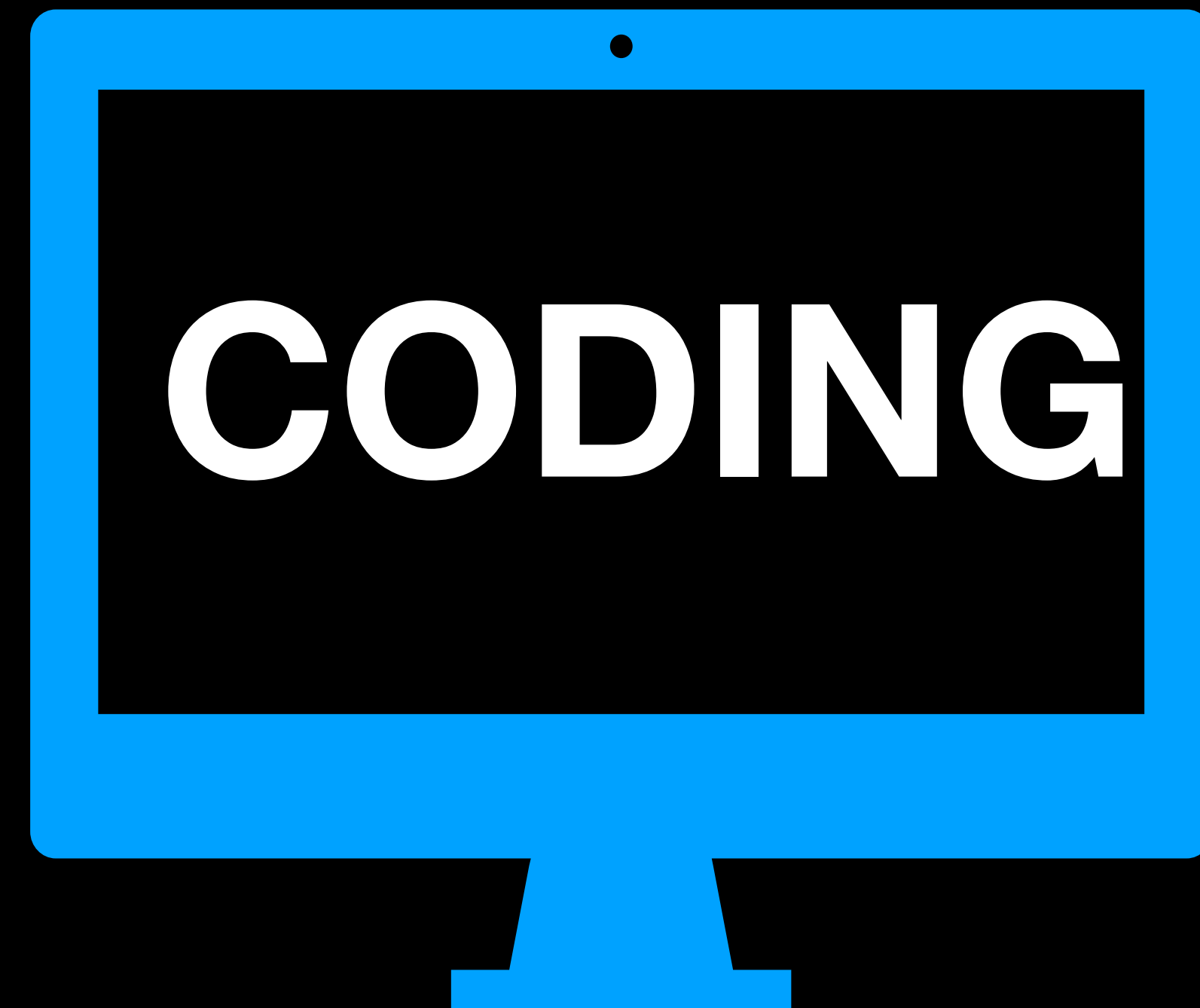
Where,

- ▲ `encoding(optional)` :Encoding standard to use. Default is UTF-8

- ▲ `error(Optional)` : Response when encoding fails. Six types of error response
Default is “strict”

- Output : It returns the new encoded string

- Python String Methods
 - `str.encode(encoding, error)`



- Python String Methods

- `str.endswith(suffix, start, end)`

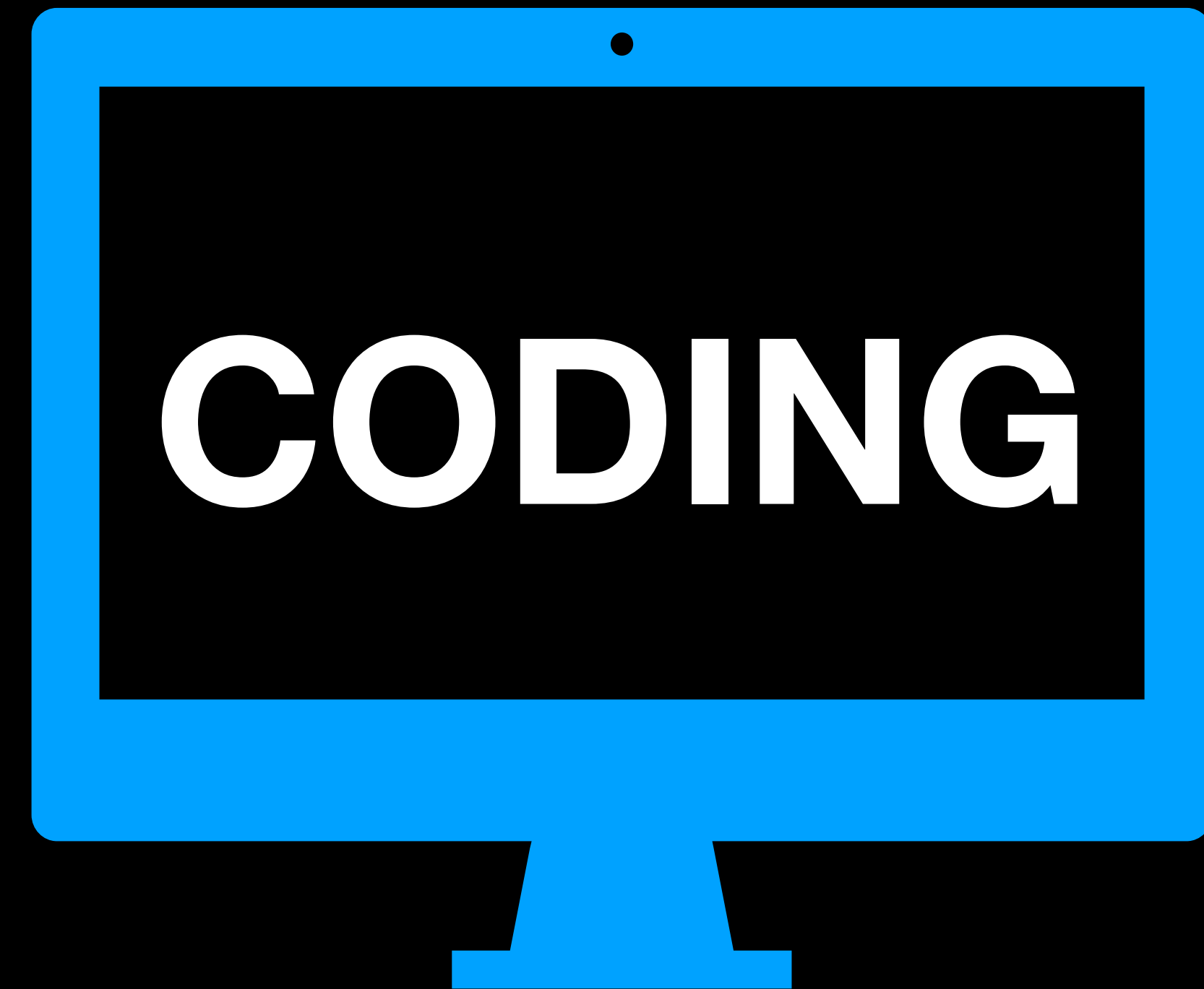
- ▲ `value(Required)` : String that needs to be checked

- ▲ `start(Optional)` : Start position from where string need to be searched

- ▲ `end(Optional)` : End position from where string need to be searched

- **Output** : It returns the True if Sub-String found else False

- Python String Methods
 - `str.endswith(suffix, start, end)`



Next

- String Methods...

expandtabs()
find()
format()
format_map()



Thank You
Happy Learning
Keep Watching