YAML - SCALARS

Yaml Breakdown

- Mapping
- Sequences
- **Scalars**
- Structures
- Comments
- Tags
- Anchors

Scalars

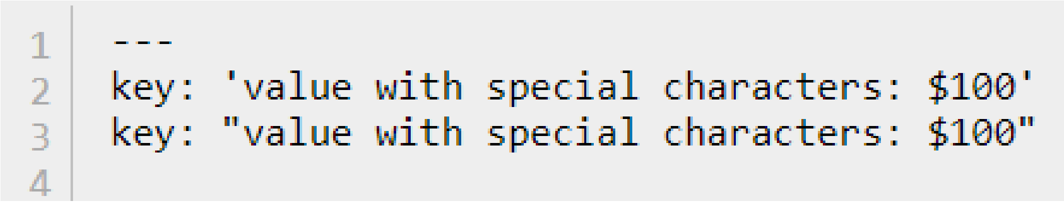
- Single values in YAML.
- Handling numbers, strings, and Booleans.
- Special characters and escaping.

What Are Scalars?

- A scalar is a single unit of data, representing a single value.
- Scalars are the simplest and most basic data type in YAML.
- They can be of various types, including strings, numbers, Booleans, null, and special types like timestamps.
- Scalars can be represented in different styles, such as plain, single-quoted, double-quoted, folded, and literal.

Single-Quoted Scalars

- Single-Quoted scalars encloses the scalars in single quote in which special characters are escaped and variables are not expanded.

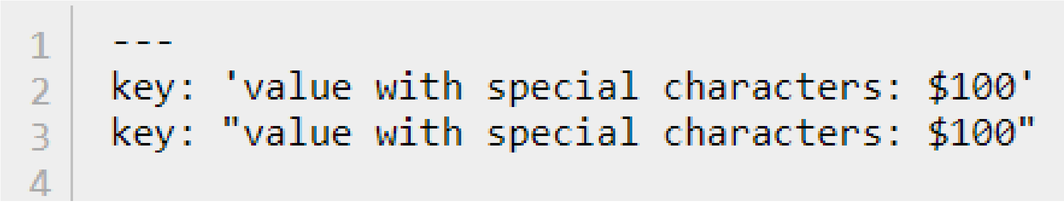


```
---
key: 'value with special characters: $100'
key: "value with special characters: $100"

```

Double-Quoted Scalars

- Double-Quoted scalars are like single quoted, but they are enclosed in double quotes and here it allows special characters escaping and variable expansion.

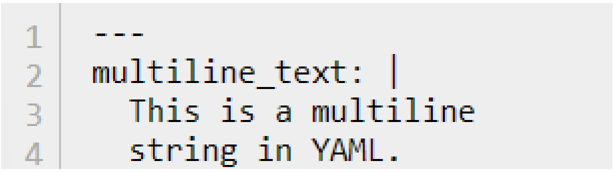


```
---
key: 'value with special characters: $100'
key: "value with special characters: $100"

```

Literal Block Scalars

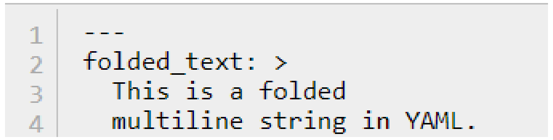
- In literal block styles it preserves newlines and indentation.
- It uses ‘|’ to indicate the literal style.



```
---
multiline_text: |
  This is a multiline
  string in YAML.
```

Folded Block Scalars

- In the folded block style, it preserves newlines but not indentation
- It uses ‘**>**’ to indicate folded style



```
---
folded_text: >
  This is a folded
  multiline string in YAML.
```

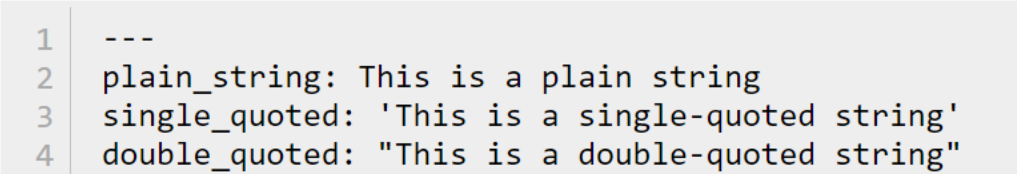
Types Of Scalars:

- Strings.
- Numbers.
- Booleans.
- Null.
- Special Types.
- Timestamps.

Examples (1-3):

- Strings:
  - A sequence of characters, typically representing text.

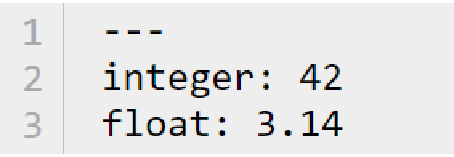
  Example:



```yaml
---
plain_string: This is a plain string
single_quoted: 'This is a single-quoted string'
double_quoted: "This is a double-quoted string"
```

- Numbers:
  - Numeric values, including integers and floating-point numbers.

  Example:

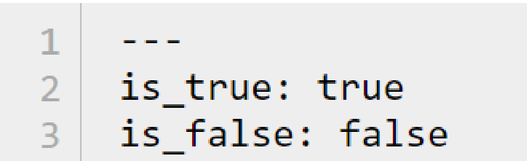


```yaml
---
integer: 42
float: 3.14
```

Examples (2-3):

- Booleans:
  - Represent true or false.

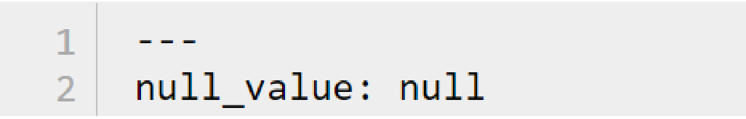
  Example:



```
---
is_true: true
is_false: false
```

- Null:
  - Represents null or empty values.

  Example:

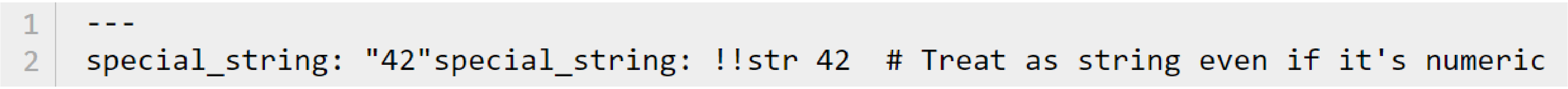


```
---
null_value: null
```

Examples (3-3):

- Special Types:
  - Special types include ‘!!str’ for strings, ‘!!int’ for integers, ‘!!float’ for floating-point numbers, etc..

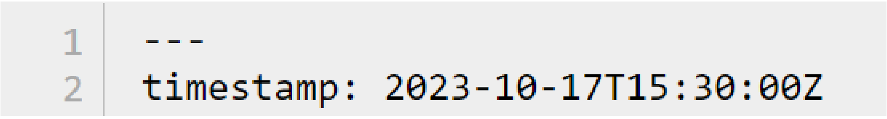
  Example:



```
---
special_string: "42"special_string: !!str 42  # Treat as string even if it's numeric
```

- Timestamps:
  - Representing specific points in time.

  Example:

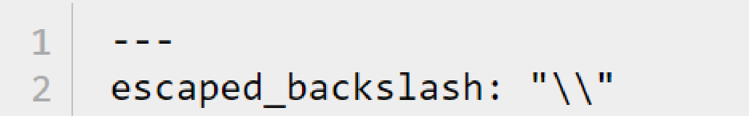


```
---
timestamp: 2023-10-17T15:30:00Z
```

Special Character And Escaping.

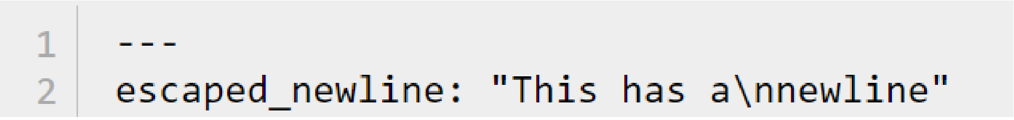
- Special characters have specific meanings and can affect the interpretation of the data.
- To represent these characters as literal values (without special meanings), you can use escaping techniques. Escaping involves preceding a special character with a backslash (\).

Examples: Special Characters And Escaping



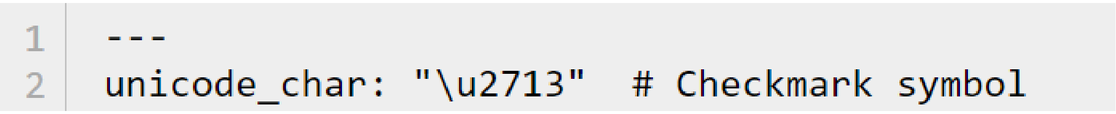
```
---
escaped_backslash: "\\"
```

To Repeat Backslash Itself.



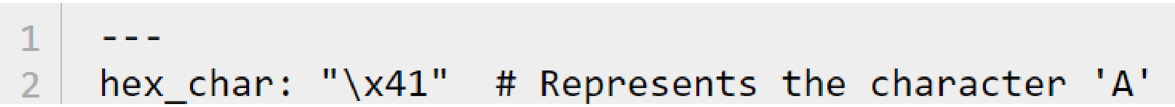
```
---
escaped_newline: "This has a\nnewline"
```

YAML Supports escape sequences like \n (new line), \t (Tab), etc.



```
---
unicode_char: "\u2713"  # Checkmark symbol
```

To Represent Unicode characters using their Unicode points.



```
---
hex_char: "\x41"  # Represents the character 'A'
```

To Represent characters using their Hexadecimal representation