

- Tuple
- Tuple has only two methods:
 - a. `index(x)`
 - b. `count(x)`
- tuple being immutable, we cannot modify the contents, but we can replace all the contents of tuple i.e. re-tupling is allowed
- a list within a tuple is still dynamic. It'll not leave its property being dynamic even if it is a part of tuple
- 3d data has time as 3rd dimension
- types of data:
 - a. Temporal: cannot exist without time, music, video, etc.
 - b. Non-temporal: can exist without time, picture, text, etc.

- Set targets
 - a. Unique
 - b. Custom index
 - c. Heterogeneity
 - d. Dynamic behavior
 - e. Not indexable/subscriptable
 - f. Supports auto-indexing/auto-iteration
- discrete mathematics:
 - a. Union
 - b. Intersection
 - c. Symmetric difference ($\text{union}(A, B) - \text{intersection}(A, B)$)
 - d. Difference ($A-B$ or $B-A$)
- financial companies put lot of set questions
- methods in set are:
 - a. `add(data)`: adds one item to a set
 - b. `update(set)`: adds set, basically union
 - c. `remove(data)`: removes data, throws error, if data is not found
 - d. `discard(data)`: removes data without error
- `s.pop()` without index pops element from front in set

- dict (dictionary)
 - a. Follows key-value pair
 - b. Heterogeneous
 - c. Key must be unique, value may not be unique
 - d. Onwards python v3.7, items are ordered
- dict has following methods:
 - a. `var.keys()`: returns list of all keys
 - b. `var.values()`: returns list of all values
 - c. `var.items()`: returns list of tuples of key-value pairs
 - d. `var.update({key:value})`:

- e. `var.pop()`:
- f. `Del var[key]`:
- g. `var.clear()`:

→ by default, if dict is given to auto-iterator, only keys are printed

→ String

- a. Immutable
- b. `s`

→ string has following operators:

- a. `+`
- b. `*`
- c. `[]`
- d. `==`
- e. `!=`
- f. `in`
- g. `not in`

→ string has methods:

- a. `capitalize()`: capitalizes first letter of first word only
- b. `title()`: capitalizes first letter of each word, title-case
- c. `upper()`: converts lowercase to uppercase
- d. `lower()`: converts uppercase to lowercase
- e. `swapcase()`: inverts case for all letters in a string
- f. `count()`: returns count of sub-str
- g. `find()`: returns index if found, goes from start to end, +ve index
- h. `rfind()`: reverse find, goes from end to start, +ve index
- i. `index()`: returns index if found, but throws exception if not found
- j. `center()`:
- k. `ljust()`:
- l. `rjust()`:
- m. `lstrip()`:
- n. `rstrip()`:
- o. `strip()`:
- p. `replace(old, new [, max])`
- q. `split()`: by default, it tokenizes on basis of blank spaces; returns a list of tokens
- r. `splitlines()`:

→ `in` operator and `not in` operator are case-sensitive

→ when string is given to `set()`, it is broken in character sequence, unique, custom order

→ always try to search words in a list, and not in a normal string, as normal string would trap sub string too

→ iterator, comparator, decorator