Computer Code for Beginners
Week 5

Matt Luckcuck

19th October 2017

Last Time

Previously...

- Functions
- Sequences
  - List
  - String
  - Tuple

Outline

This Week. . .

- Sequences Recap
- Dictionaries

Sequence Recap

Sequences

Sequences Recap

- A sequence is an ordered set of data
- Zero-Indexed
- List
  - `colours = ["Red", "Blue", "Green"]`
  - Mutable
- String
  - `s = "Purple"`
  - Immutable
- Tuple
  - `t = (170,52)`
  - Immutable

Sequences

String

- Another sequence type
  - Each character is indexed by a number
- `s = "Purple"`
- Allows some sequence operations:
  - Indexing – `s[0]`
    - `"P"`
  - Slicing – `s[0:2]`
    - `"Pu"`
- But Strings are *immutable*

Sequences

Tuple

- Another ordered set of data
- Cannot be changed (immutable)
  - We say this is *immutable*
- Zero-indexed like a list or string
- `t = (170,52)` Simple Tuple (Pair)
- `t[0]` is
  - 170
- `t[1]` is
  - 52
- But Tuples are *immutable*

Sequences

Common Sequence Operations

- Indexing: `s[0]`
- Slicing: `s[0:2]`
- Length: `len(s)`
- Member of: `42 in s`
- Not a Member of: `42 not in s`

Dictionaries

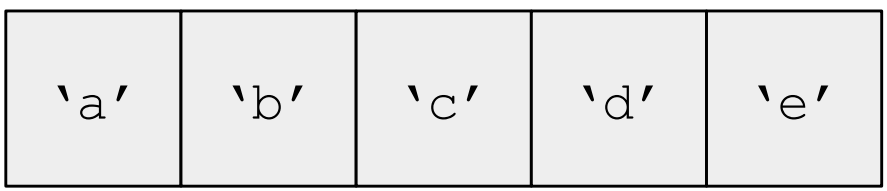
Dictionaries

Dictionaries $\neq$ Sequences

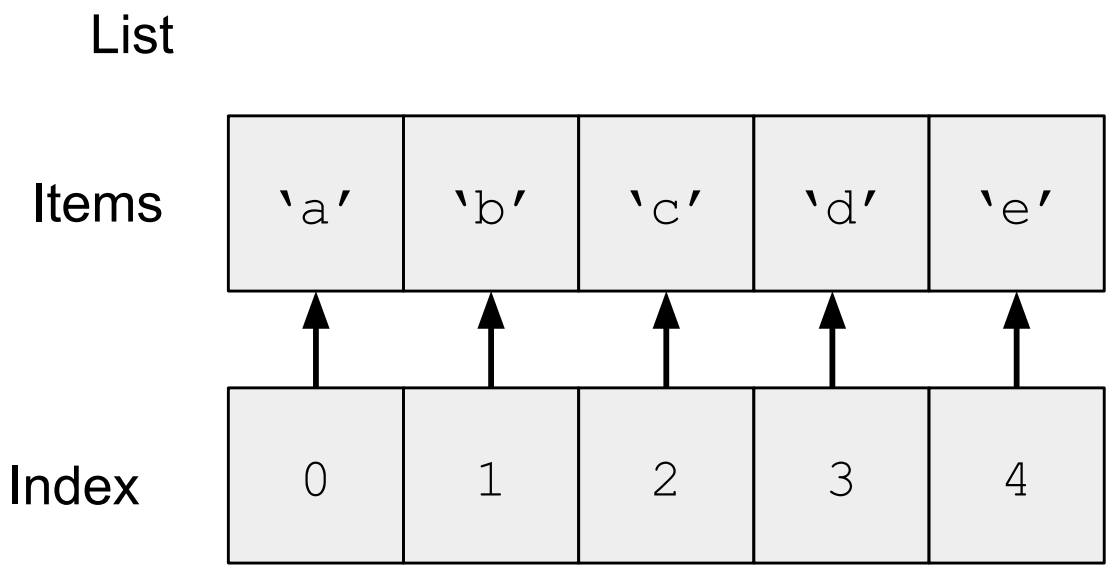
- Unordered set of data
- `d = {"Octopus":42,"cat": 5}`
  - Sometimes called *Maps* or *Associative Arrays*
- Maps `keys` to `values`
  - Think of two lists, or
  - Think of a set of pairs (Key, Value)
- Each item is a key and a value

Dictionaries

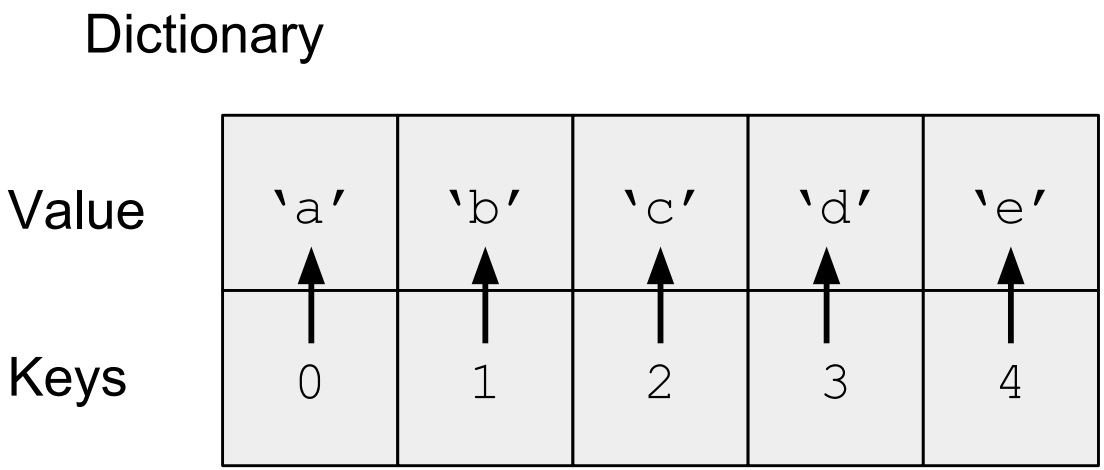
List



Dictionaries



Dictionaries



Dictionaries

Dictionary Keys

- Keys can be any immutable type:
  - String
  - Number
  - Tuples (only containing Strings, Numbers, or Tuples)
- Keys must be unique in one Dictionary

Dictionaries

Useful Dictionary Operations

- `d = dict()` or `d = {}` makes a new empty dictionary
- `d = {"a":7, "b":12}` makes a dictionary with entries
  - or `d = dict([("a", 7), ("b", 12)])`
- `d["b"]` gives us 7
- `d["c"] = 5` adding an entry

Dictionaries

Useful Dictionary Operations

```
d = {"a":7, "b":12}
```

- `d.keys()` Lists the keys
- `d.values()` Lists the values
- `d.items()` lists tuples of key, value pairs

Dictionaries

Useful Dictionary Operations

```
d = {"a":7, "b":12}
```

- `d.keys()` Lists the keys
  - `["a", "b"]`
- `d.values()` Lists the values
- `d.items()` lists tuples of key, value pairs

Dictionaries

Useful Dictionary Operations

```
d = {"a":7, "b":12}
```

- `d.keys()` Lists the keys
  - `["a", "b"]`
- `d.values()` Lists the values
  - `[7, 12]`
- `d.items()` lists tuples of key, value pairs

Dictionaries

Useful Dictionary Operations

```
d = {"a":7, "b":12}
```

- `d.keys()` Lists the keys
  - `["a", "b"]`
- `d.values()` Lists the values
  - `[7, 12]`
- `d.items()` lists tuples of key, value pairs
  - `[("a", 7), ("b", 12)]`
- Order unreliable

Summary

Summary

Summary

- Dictionaries
  - New compound data type
  - Map keys to values

Summary

Exercises

- Letters in a String
  - Using a dictionary to keep track of how many of each different letter there are in a string
- Morse Code
  - Dictionary of letters mapping to their Morse code representation
  - Converting between text and Morse using the dictionary
  - Reversing the dictionary to reverse the translation