A Python dictionary (dict) is a special container type. It contains a collection of items, which are called key-value pairs and have the following form:

\[ \text{key:val} \]

The key in an item must be an immutable object and the val can be any type of object. The items contained in a dictionary are delimited by curly braces ({ and }) and separated by commas. For example,

```
A = {'CA': 38332521, 'TX': 26448193, 'MI': 9895622}
```

creates a new dict containing 3 items and assigns variable A a reference to it.

The keys in a dictionary are used to retrieve and update values, and to create items.

- To create an item or update a value for a key: \( \text{a_dict[k]} = \text{exp} \)
  - If \( \text{a_dict} \) contains an item whose key equals \( k \), then the assignment replaces the value in this item with the value of \( \text{exp} \); otherwise, the assignment creates an item with key equal to \( k \) and value equal to \( \text{exp} \) and adds this new item to \( \text{a_dict} \).

- To retrieve a value: when not on the left-side of an assignment, \( \text{a_dict[k]} \) returns the value in the item in \( \text{a_dict} \) whose key equals \( k \); or raises an error, if \( \text{a_dict} \) does not contain any item whose key equals \( k \).

A dict is iterable; but you iterate through a dict using its keys.

```
\text{exp in a_dict}: \text{returns True if exp is a key in a_dict; and False, otherwise.}
```

```
\text{for var in a_dict: iterates through the keys of a_dict, assigning each key to var in its turn and executing the associated suite}
```

```
\text{len(a_dict): returns the number of items in a_dict}
\text{max(a_dict): returns the maximum key in a_dict}
\text{min(a_dict): returns the minimum key in a_dict}
```

```
\text{a_dict.keys(): returns the collection (iterable) of keys in a_dict}
\text{a_dict.values(): returns the collection (iterable) of values in a_dict}
\text{a_dict.items(): returns the collection (iterable) of items in a_dict}
\text{del a_dict[exp]: deletes the item in a_dict whose key equals exp}
```