**Словари: Dict**

КЛЮЧ:ЗНАЧЕНИЕ

|  |  |
| --- | --- |
| **massiv\_dict= {"Dima":53827,"Sveta":75095}**  **massiv\_dict1=dict(Artem=99544,Vika=14789)** | Обявление словаря |
| **massiv\_dict["Dima"]**  **massiv\_dict1.get("Artem", 0)**  **massiv\_dict1.get("Artem")**  None | Выбор ЗНАЧЕНИЯ по ключу. Если ключ артем отсуцтвует, возвращает для него значение по умолчанию=0. Новый єлемент не создает!!! |
| **massiv\_dict["Polya"]= "n/a"** | Добавляет новые элементы в словарь. По ключевому значению изменяет значение старых элементов |
| **del massiv\_dict["Name"]** | Удаляет элементы по ключу |
| **massiv\_dict.keys()**  **massiv\_dict.values()**  **massiv\_dict.items()** | Возвращает общий список КЛЮЧЕЙ / ЗНАЧЕНИЙ. В последнем случае И ключей и значений |
| **sorted(massiv\_dict.keys())**  **sorted(massiv\_dict.values())** | Сортировка всех  КЛЮЧЕЙ / ЗНАЧЕНИЙ |
| **print("Dima" in massiv\_dict)**  **print("Vika" not in massiv\_dict)** | Проверка на существование либо отсуцтвие значения в словаре. Возвращает Булиан |
| **for key, val in massiv\_dict.items():                      print(key,val)** | Проход циклом по словарю либо спису |
| **massiv\_dict.pop("Dima")** | Удаляет по ключу элемент. Возвращает его в переменной |
| **massiv\_dict.popitem()** | Удаляет последний элемент |
| **massiv\_dict.setdefault("Potalok")** | Устанавливает значение None заданному новому ключу. |
| **massiv\_dict.setdefault("Potalok",”Value”)** | Устанавливает значение заданному новому ключу. Если ключ уже присуцтвует и значение его не совпадает с функуией. То исходное значение в словаре не перезаписывается |
| **D = dict(zip([‘a’, ‘b’, ‘c’], [1, 2, 3]))**  {‘a’: 1, ‘c’: 3, ‘b’: 2} | *Генератор словарей из списка ключей и списка значений* |
| **D = {k: v for (k, v) in zip([‘a’, ‘b’, ‘c’], [1, 2, 3])}** | Генератор словарей. Выражение **k: v для переменных (k, v) из списков … …** |
| **D = {x: x \*\* 2 for x in [1, 2, 3, 4]}**  {1: 1, 2: 4, 3: 9, 4: 16} | Генератор словаря квадратов чисел |
| **D = {c: c \* 4 for c in ‘SPAM’}**  {‘A’: ‘AAAA’, ‘P’: ‘PPPP’, ‘S’: ‘SSSS’, ‘M’: ‘MMMM’} |  |
| **D = {c.lower(): c + ‘!’ for c in [‘SPAM’, ‘EGGS’, ‘HAM’]}**  {‘eggs’: ‘EGGS!’, ‘ham’: ‘HAM!’, ‘spam’: ‘SPAM!’} |  |
| **D = dict.fromkeys([‘a’, ‘b’, ‘c’], 0)**  **D = {k:0 for k in [‘a’, ‘b’, ‘c’]}**  {‘a’: 0, ‘c’: 0, ‘b’: 0} | Инициализация словаря из списка 0-мя |
| **D = dict.fromkeys(‘spam’)**  **D = {k: None for k in ‘spam’}**  {‘a’: None, ‘p’: None, ‘s’: None, ‘m’: None} |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

*Таблица 8.2. Литералы словарей и операции*

|  |  |
| --- | --- |
| **Операция** | **Интерпретация** |
| D = {} | Пустой словарь |
| D = {‘spam’: 2, ‘eggs’: 3} | Словарь из двух элементов |
| D = {‘food’: {‘ham’: 1, ‘egg’: 2}} | Вложение |
| D = dict(name=’Bob’, age=40)  D = dict(zip(keyslist, valslist))  D = dict.fromkeys([‘a’, ‘b’]) | Альтернативные способы создания словарей:  именованные аргументы, применение функции zip, списки ключей |
| D[‘eggs’]  D[‘food’][‘ham’] | Доступ к элементу по ключу |
| ‘eggs’ in D | Проверка на вхождение: проверка нали­чия ключа |
| D.keys()  D.values()  D.items()  D.copy()  D.get(key, default)  D.update(D2)  D.pop(key) | Методы: список ключей,  список значений,  список ключей и значений,  копирование,  получение значения по умолчанию,  слияние,  удаление и так далее |
| len(D) | Длина (количество элементов) |
| D[key] = 42  del D[key] | Добавление/изменение ключей, удаление ключей |
| list(D.keys())  D1.keys() & D2.keys() | Представления словарей (в Python 3.0) |
| D = {x: x\*2 for x in range(10)} | Генераторы словарей (в Python 3.0) |