**1 ПРАКТИКА**

|  |  |
| --- | --- |
| 1. В программе уже задан словарь с именами питомцев. Напиши программу, которая выведет только уникальные значения словаря в порядке возрастания.   my\_dict = {1:'Пушок', 2: 'Гарфильд', 3: 'Черныш', 4:'Марс', 5: 'Снежок', 6: 'Пушок', 7: 'Марс'} | **Входные данные**  **Выходные данные**  ['Гарфильд', 'Марс', 'Пушок', 'Снежок', 'Черныш'] |
| 1. Напиши программу, которая запрашивает у пользователя строку, а потом определяет, все ли символы в строке уникальные. Если да, то выводится сообщение “Все символы уникальные”, иначе сообщение “Нет”. | **Входные данные**  Я приеду  **Выходные данные**  Все символы уникальные |
| 1. Пользователь вводит элементы списка через запятую. Напиши функцию, которая принимает список, удаляет все повторения элементов и выводит новый список с уникальными элементами. Список должен быть отсортирован по возрастанию. Обязательно используй множества для решения этой задачи. | **Входные данные**  5,7,8,5,1,3,1,6 **Выходные данные**  ['1', '3', '5', '6', '7', '8'] |

**2 ПРАКТИКА**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1. В программе задан список. Создай множество из списка. Напиши программу, которая предлагает выбор действий пользователю: 1- Добавить, 2-Удалить, 3-Выйти.   Если пользователь выбирает “Добавить”, то программа запрашивает элемент и добавляет в множество. Программа снова предлагает действия.  Если пользователь выбирает “Удалить”, то программа запрашивает элемент и удаляет, если он есть. Если элемента нет, то программа ничего не делает. Программа снова предлагает действия.  Когда пользователь нажимает “3”, программа выводит полученное множество.  p = [5, 12, 18, 46, 34, 6] | | **Входные данные**  1  56  1  12  2  34  3  **Выходные данные**  {5, 6, 12, 46, 18, 56} | | | |
| 1. Ученики 9 классов уже предложили свои варианты, где отпраздновать выпускной. Все названия собраны в множество s = {'школа', 'парк', 'теплоход', 'кинотеатр'}. Теперь свои варианты будут предлагать ученики 11 классов.   Напиши программу, которая запрашивает названия мест через пробел одной строкой, а потом выводит отсортированный список по возрастанию, в котором названия мест не повторяются. | | **Входные данные**  кафе парк  **Выходные данные**  ['кафе', 'кинотеатр', 'парк', 'теплоход', 'школа'] | | | |
| 1. Напиши программу, которая запрашивает числа, пока пользователь не введет 0. Все числа добавляются в множество s = {5, 24, -5, 0, 667, -3, -6}. Потом программа выводит список без отрицательных чисел, отсортированный по убыванию. | **Входные данные**  -12  7  45  -34  0 | | **Выходные данные**  [667, 45, 24, 7, 5] | |
| 1. Пользователь вводит строку, в которой записана последовательность целых чисел через пробел. Для каждого числа нужно вывести “Нет”, если число не встречалось в последовательности ранее или вывести “Повтор”, если это число уже встречалось. | **Входные данные**  56 4 3 5 4 56 7 8  **Выходные данные**  Нет  Нет  Нет  Нет  Повтор  Повтор  Нет  Нет | | |

**1 ПРАКТИКА**

|  |  |
| --- | --- |
| 1. В программе уже задан словарь с именами питомцев. Напиши программу, которая выведет только уникальные значения словаря в порядке возрастания.   my\_dict = {1:'Пушок', 2: 'Гарфильд', 3: 'Черныш', 4:'Марс', 5: 'Снежок', 6: 'Пушок', 7: 'Марс'} | **Входные данные**  **Выходные данные**  ['Гарфильд', 'Марс', 'Пушок', 'Снежок', 'Черныш'] |
| 1. Напиши программу, которая запрашивает у пользователя строку, а потом определяет, все ли символы в строке уникальные. Если да, то выводится сообщение “Все символы уникальные”, иначе сообщение “Нет”. | **Входные данные**  Я приеду  **Выходные данные**  Все символы уникальные |
| 1. Пользователь вводит элементы списка через запятую. Напиши функцию, которая принимает список, удаляет все повторения элементов и выводит новый список с уникальными элементами. Список должен быть отсортирован по возрастанию. Обязательно используй множества для решения этой задачи. | **Входные данные**  5,7,8,5,1,3,1,6 **Выходные данные**  ['1', '3', '5', '6', '7', '8'] |

**2 ПРАКТИКА**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1. В программе задан список. Создай множество из списка. Напиши программу, которая предлагает выбор действий пользователю: 1- Добавить, 2-Удалить, 3-Выйти.   Если пользователь выбирает “Добавить”, то программа запрашивает элемент и добавляет в множество. Программа снова предлагает действия.  Если пользователь выбирает “Удалить”, то программа запрашивает элемент и удаляет, если он есть. Если элемента нет, то программа ничего не делает. Программа снова предлагает действия.  Когда пользователь нажимает “3”, программа выводит полученное множество.  p = [5, 12, 18, 46, 34, 6] | | **Входные данные**  1  56  1  12  2  34  3  **Выходные данные**  {5, 6, 12, 46, 18, 56} | | | |
| 1. Ученики 9 классов уже предложили свои варианты, где отпраздновать выпускной. Все названия собраны в множество s = {'школа', 'парк', 'теплоход', 'кинотеатр'}. Теперь свои варианты будут предлагать ученики 11 классов.   Напиши программу, которая запрашивает названия мест через пробел одной строкой, а потом выводит отсортированный список по возрастанию, в котором названия мест не повторяются. | | **Входные данные**  кафе парк  **Выходные данные**  ['кафе', 'кинотеатр', 'парк', 'теплоход', 'школа'] | | | |
| 1. Напиши программу, которая запрашивает числа, пока пользователь не введет 0. Все числа добавляются в множество s = {5, 24, -5, 0, 667, -3, -6}. Потом программа выводит список без отрицательных чисел, отсортированный по убыванию. | **Входные данные**  -12  7  45  -34  0 | | **Выходные данные**  [667, 45, 24, 7, 5] | |
| 1. Пользователь вводит строку, в которой записана последовательность целых чисел через пробел. Для каждого числа нужно вывести “Нет”, если число не встречалось в последовательности ранее или вывести “Повтор”, если это число уже встречалось. | **Входные данные**  56 4 3 5 4 56 7 8  **Выходные данные**  Нет  Нет  Нет  Нет  Повтор  Повтор  Нет  Нет | | |