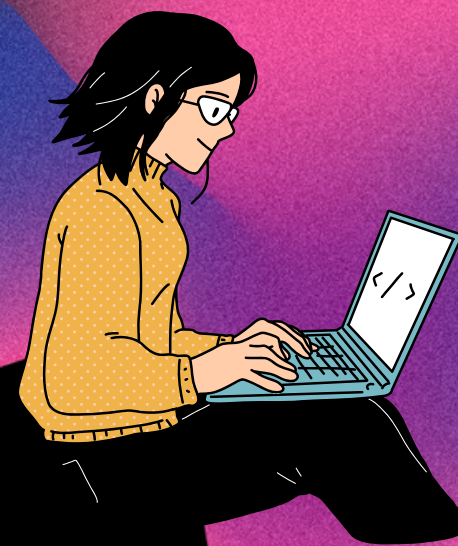
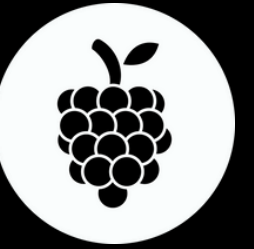


ПРАКТИЧЕСКАЯ РАБОТА 6 – МНОЖЕСТВА



TANKURAI
CEO NARIMAN



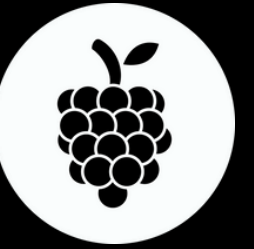
```
# Создаем множество из 5 элементов
my_set = {12, 67, 45, 89, 33}
print("Исходное множество:", my_set)

# Удаляем элемент со значением 67, если он есть в множестве
if 67 in my_set:
    my_set.remove(67)

# Добавляем два новых элемента
my_set.add(55)
my_set.add(71)

# Выводим измененное множество
print("Измененное множество:", my_set)
```





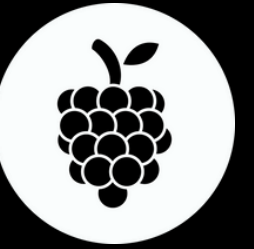
```
my_list = [1, 53, 8, 9, 34, 1, 0, 53, 53, 8, 73, 5]

# Преобразовываем список во множество (set) для удаления дубликатов
my_set = set(my_list)

# Преобразовываем множество обратно в список
my_list_without_duplicates = list(my_set)

print("Исходный список:", my_list)
print("Список без повторяющихся элементов:", my_list_without_duplicates)
```



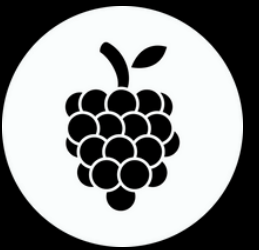


```
my_set = set([1, 2, 3, 4, 5])
```

```
my_set = {1, 2, 3, 4, 5}
```

```
my_set = set()
for i in range(1, 6):
    my_set.add(i)
```





```
# Создаем обычное множество (set)
set1 = {1, 3, 4, 6, 7}

# Создаем неизменяемое множество (frozenset)
frozenset1 = frozenset({2, 4, 7, 8, 9})

# Объединяем оба множества
union_set = set1.union(frozenset1)

print("Объединенное множество:", union_set)

# Добавляем элементы 2 и 5
union_set.add(2)
union_set.add(5)

print("Множество после добавления 2 и 5:", union_set)

# Удаляем число 2 и первый элемент (используя методы remove и pop)
if 2 in union_set:
    union_set.remove(2)

if len(union_set) > 0:
    union_set.pop()

print("Множество после удаления 2 и первого элемента:", union_set)
```

