# **Модули**

**import my\_mod** импортируем указаный встроенный модуль my\_mod из стандартной библиотеки python.

**import my\_mod as second\_name\_module**  переназвание модуля для коротшей записи, теперь нужно оборащатся через второе имя. Обращаемся по етому псевдониму

**import first\_module , second\_module** подключение сразу несколько модулей

**import first\_module as f , second\_module as s** подлючение нескольких модулей и их переназвание

**my\_mod.my\_fun( )** использование функции(метода) встроенного модуля

**from my\_mod import my\_fun** импортируем из модуля my\_mod только функцию my\_fun, остальные фукнции недоступны

**my\_fn ( )** обращаемся к единственой функции, которую взяли из модуля

**from my\_mod import my\_fun as second\_name\_function** импортируем из модуля отдельную функцию и переназываем ее

**import my\_python\_file** импортируем свой модуль

**from my\_python\_file import my\_fn** импортируем свою функцию из свого модуля

**from my\_module import \*** импортиуем весь функционал из указаного модуля в текущий файл

# **math**

**math.sqrt ( 144 )** возвращает корень из указаного числа

**math.ceil ( 5.3 )** округляет число в большую сторону

# **time**

**time.sleep ( 5 )** сотанавливает время выполнения кода на заданое количестов секунд

# **detatime**

**datetime.date.today( )** возвращает сегоднюю дату

**datetime.datetime.now().today()** или **hour( ), day()** **,month( )**  возвращает сегоднишние значения

# **sys**

**sys.path** возвращает пути директорий, где лежит текущий проект

# **os**

**os.name** возвращает текущую операционую систему пользователя (nt - windows, posix - macos)

# **platform**

**platform.system( )** возвращает текущую операционую систему пользователся (windows, macos)

# **copy**

**my\_list1 = deepcopy( my\_list2 )** копирование изменяемых обьектов второго уровня. У вложенных обьектов второго списка копируютса значенния, а не ссылка на внутрений список.

# **random**

**randim.randint( a , b )** возвращает случайное число от a(влюч) до b(включ)

# **webbrowser**

**webbrowser.open (“url\_adress”)** открывает адресс url в нашем браузере

# **accessify (pip install)**

**@private @protected** устанавливает жесткие режими доступа для методов

# **string**

**string.ascii\_letters** содержит все буквы англиского алфавита в верхнем и нижнем регистре

# **timeit**

**timeit.timeit (my\_method** или **my\_fn)** возвращает скорость выполения указаного метода или функции