



Python: быстрый старт.

Урок 2

Функция, как "черный ящик",
встроенные функции, порядок
вычисления выражений, объявление
собственных функций

Функция – «черный ящик»

Имя функции

Аргумент

abs (-9)

Возвращаемое
значение:
абсолютное
значение числа -9

Аргумент функции

-9

Функция возвращает
абсолютное значение числа.
Аргумент может быть *int* или
float.

9

Функция **pow(x, y)** возвращает значение **x** в степени **y**.

Эквивалентно записи **x**y**.

Функция **round(number)** возвращает число с плавающей точкой, округленное до 0 цифр после запятой (по умолчанию).

Может быть вызвана с двумя аргументами:

round(number[, ndigits])

где `ndigits` – число знаков после запятой.

Документация: <https://docs.python.org/3.4/library/functions.html#round>

Порядок вычисления

`pow (abs (-2), round(4.3))`

The diagram illustrates the order of evaluation for the expression `pow (abs (-2), round(4.3))` using green underlines and numbers:

- 1**: Underline under `-2` in `abs (-2)`.
- 2**: Underline under the entire `abs (-2)` sub-expression.
- 3**: Underline under `4.3` in `round(4.3)`.
- 4**: Underline under the entire `round(4.3)` sub-expression.
- 5**: Underline under the entire `pow (abs (-2), round(4.3))` expression.

Преобразование типов

int ()

Возвращает целочисленный объект, построенный из числа или строки, или 0, если аргументы не переданы.

Документация:

<https://docs.python.org/3.4/library/functions.html#int>

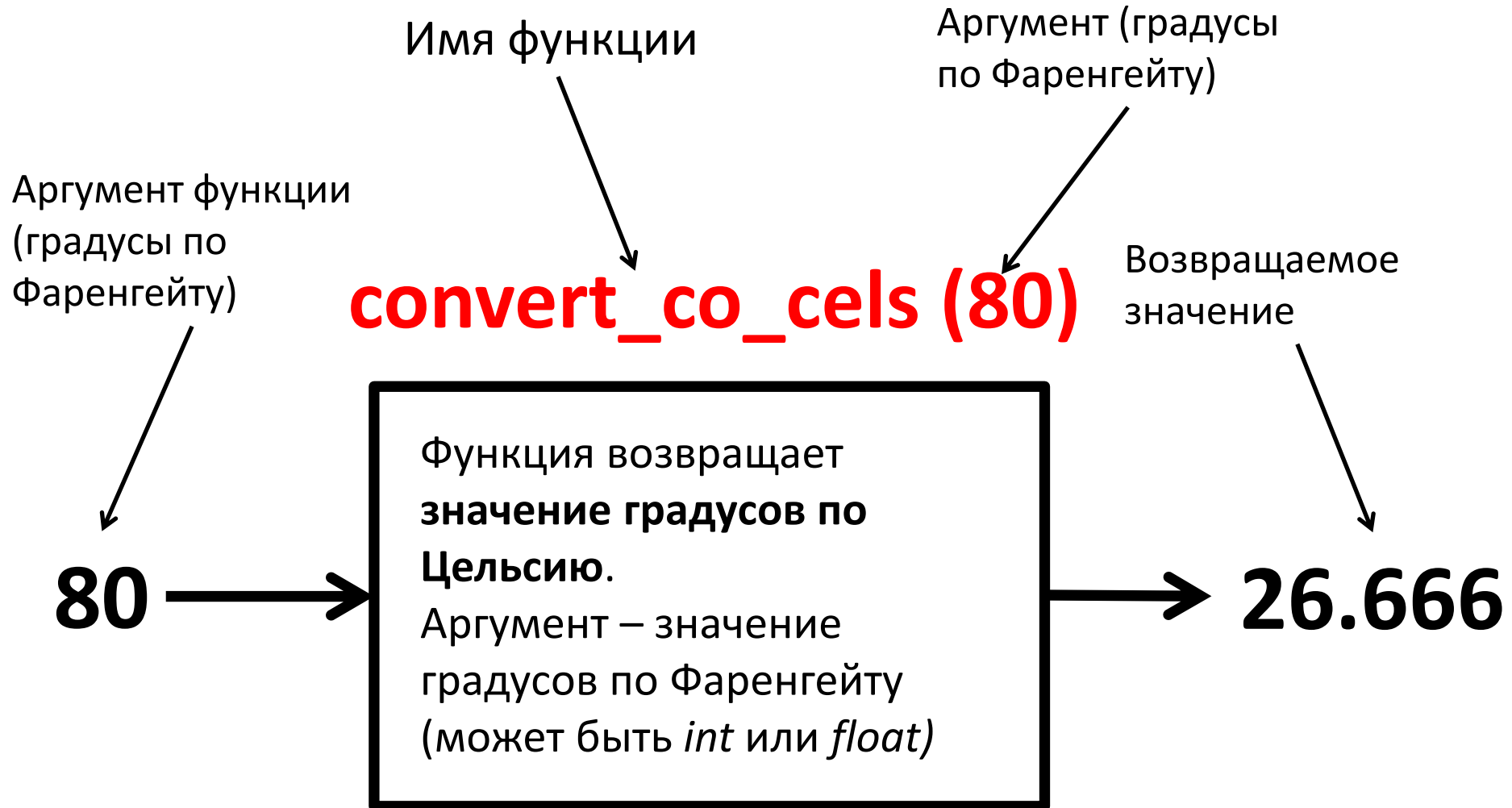
float ()

Возвращает число с плавающей точкой, построенное из числа или строки.

Документация:

<https://docs.python.org/3.4/library/functions.html#float>

Собственные функции



Собственные функции

Имя функции

Параметры (через запятую, если их несколько)

```
def convert_co_cels (fahren):  
    return (fahren - 32) * 5/9
```

Возвращаемое значение

The diagram illustrates the structure of a Python function definition. It shows the function name, parameters, and the return value, each highlighted with a different color and labeled with an arrow.

Фреймы

Здесь расположены
фреймы для
пространства имен



Объекты

Здесь расположены
объекты

Пошаговое выполнение (трассировка)



```
>>> def f(x):
```

```
...     x = 2 * x
```

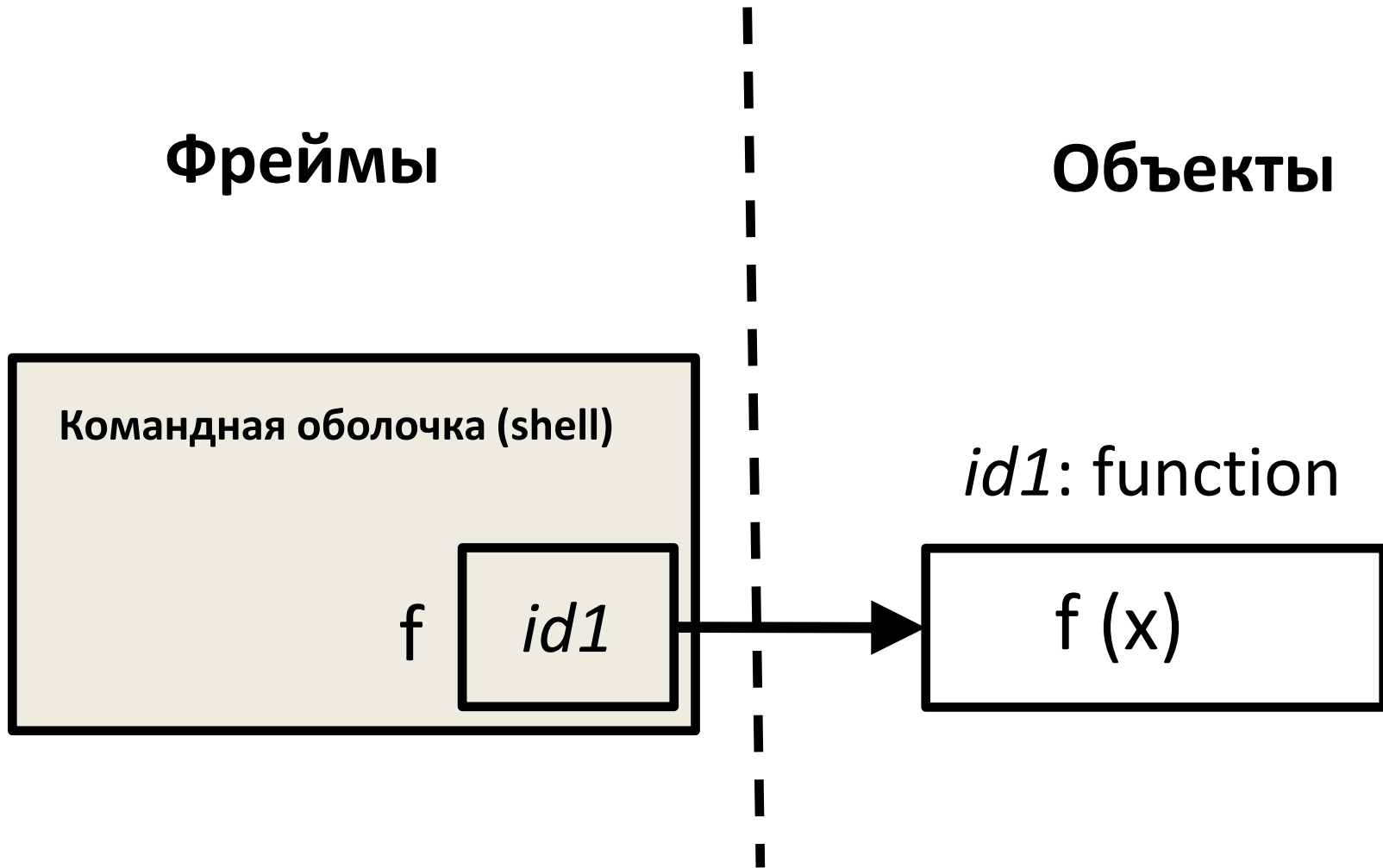
```
...     return x
```

```
>>> x = 1
```

```
>>> x = f(x + 1) + f(x+2)
```

Фреймы

Объекты



Пошаговое выполнение (трассировка)

```
>>> def f(x):
```

```
...     x = 2 * x
```

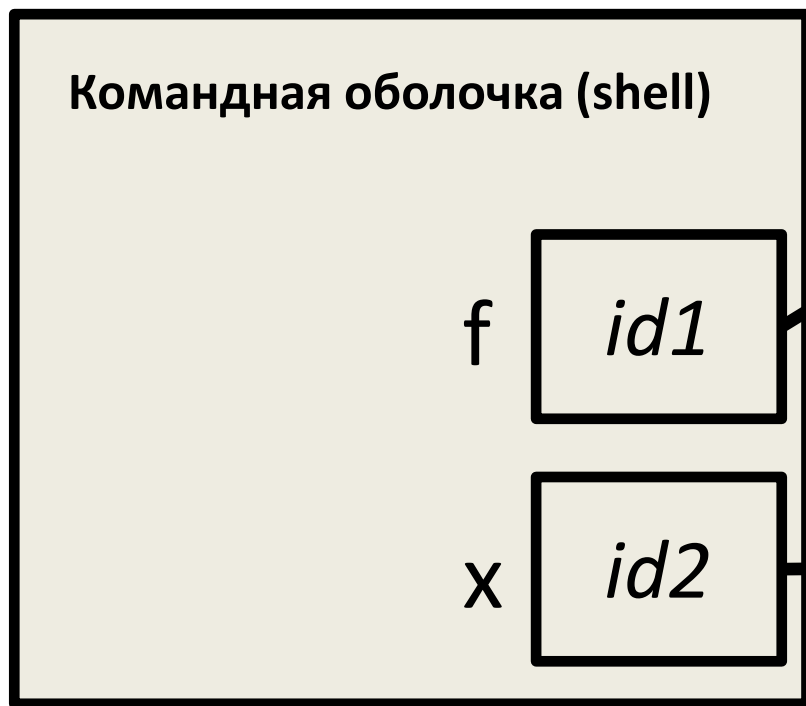
```
...     return x
```



```
>>> x = 1
```

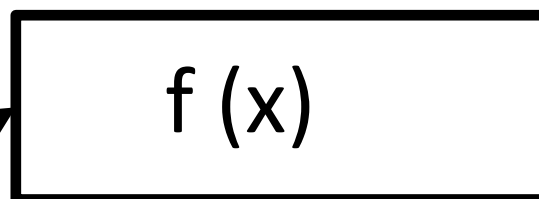
```
>>> x = f(x + 1) + f(x+2)
```

Фреймы

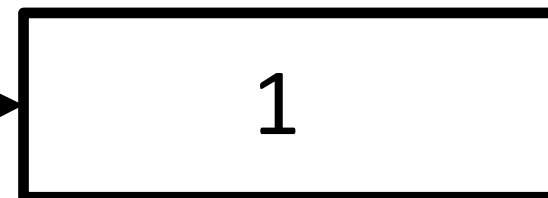


Объекты

id1: function



id2: int



Пошаговое выполнение (трассировка)

```
>>> def f(x):
```

```
...     x = 2 * x
```

```
...     return x
```

```
>>> x = 1
```

```
>>> x = f(x + 1) + f(x+2)
```



Фреймы

Командная оболочка (shell)

f

id1

x

id2

f

x

id3

Объекты

id1: function

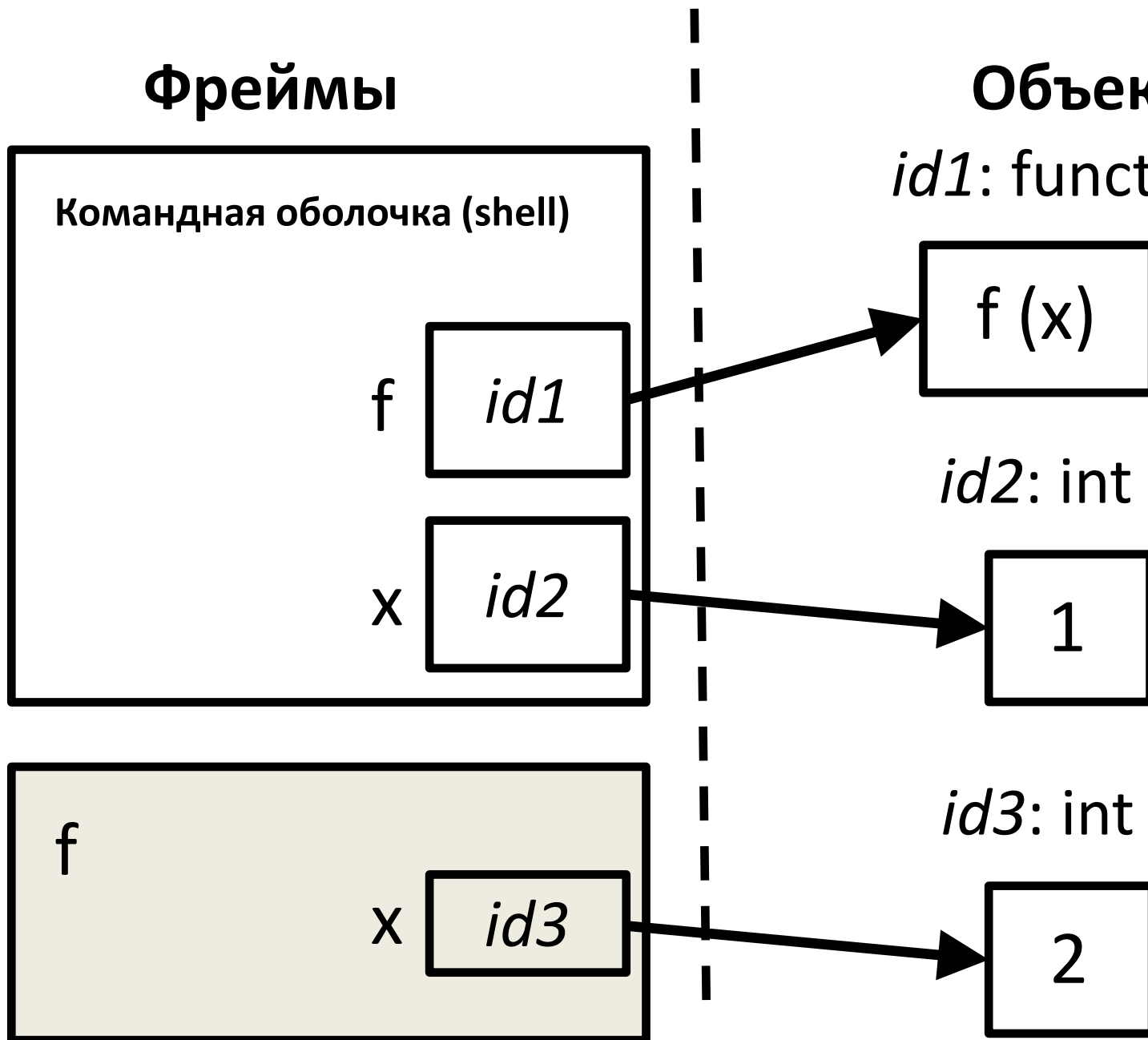
f (x)

id2: int

1

id3: int

2



Пошаговое выполнение (трассировка)

```
>>> def f(x):
```

```
...     x = 2 * x
```

```
...     return x
```

```
>>> x = 1
```

```
>>> x = f(x + 1) + f(x+2)
```


Фреймы

Командная оболочка (shell)

f

id1

x

id2

f

x

id4

Объекты

id1: function

f (x)

id2: int

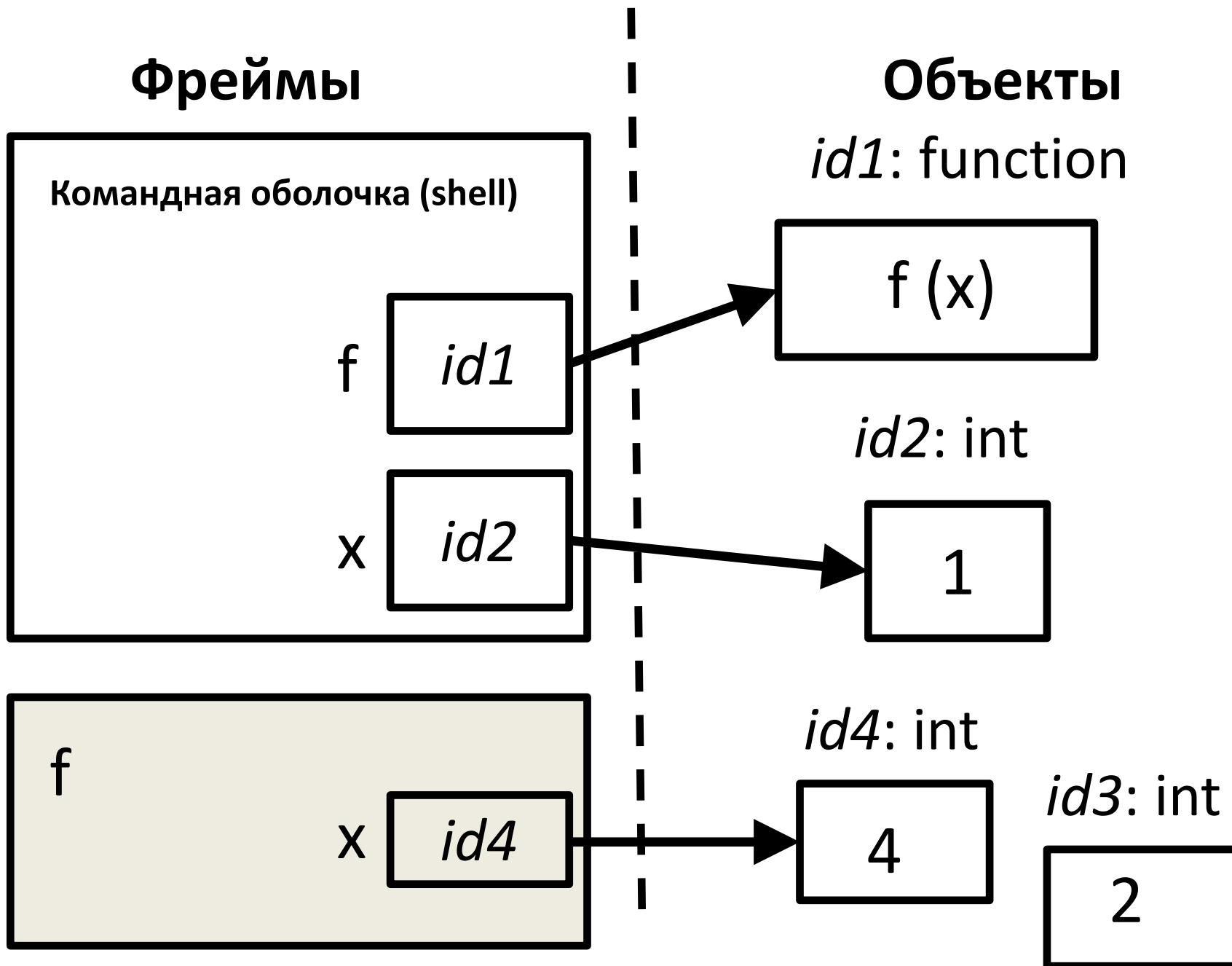
1

id4: int

4

id3: int

2



Пошаговое выполнение (трассировка)

```
>>> def f(x):
```

```
...     x = 2 * x
```

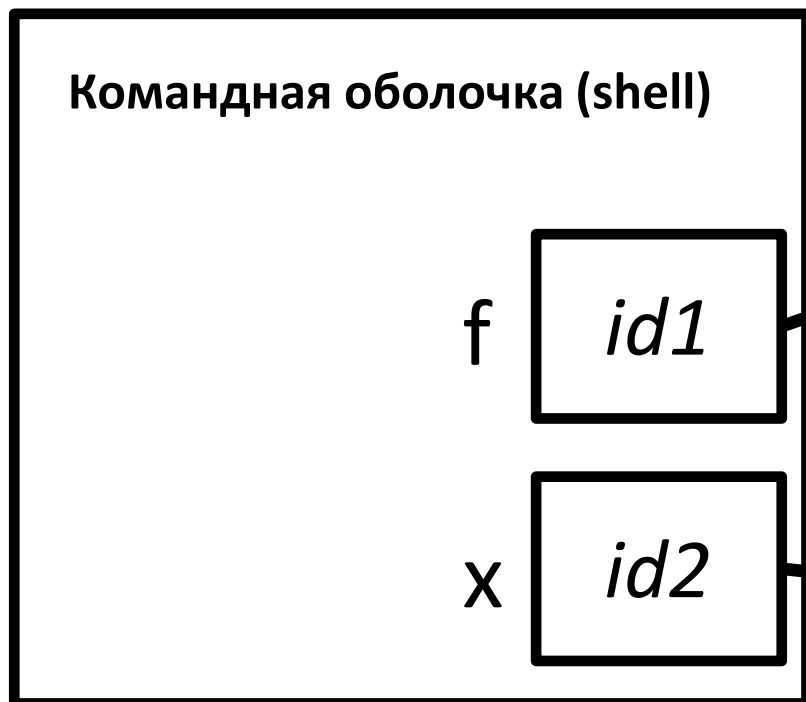
```
...     return x
```

```
>>> x = 1
```

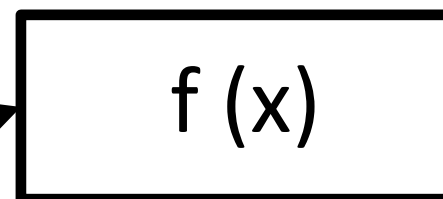
```
>>> x = f(x + 1) + f(x+2)
```

Фреймы

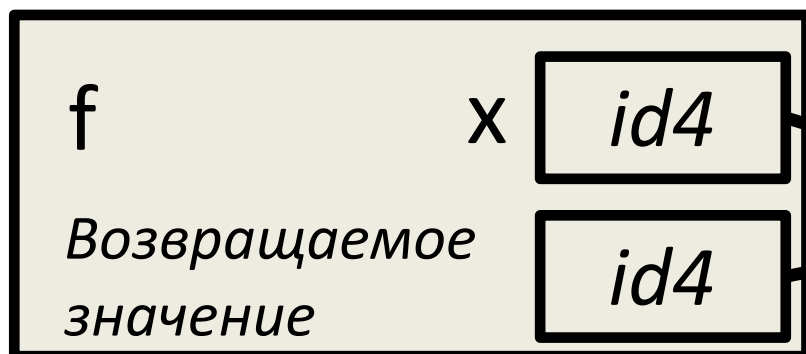
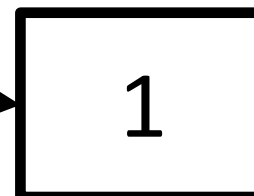
Объекты



id1: function



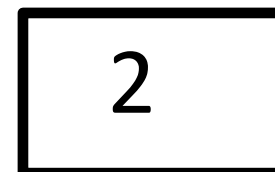
id2: int



id4: int



id3: int



Пошаговое выполнение (трассировка)

```
>>> def f(x):
```

```
...     x = 2 * x
```

```
...     return x
```

```
>>> x = 1
```

```
>>> x = f(x + 1) + f(x+2)
```



Фреймы

Объекты

Командная оболочка (shell)

f

id1

x

id2

f

x

id5

id1: function

f (x)

id2: int

1

id4: int

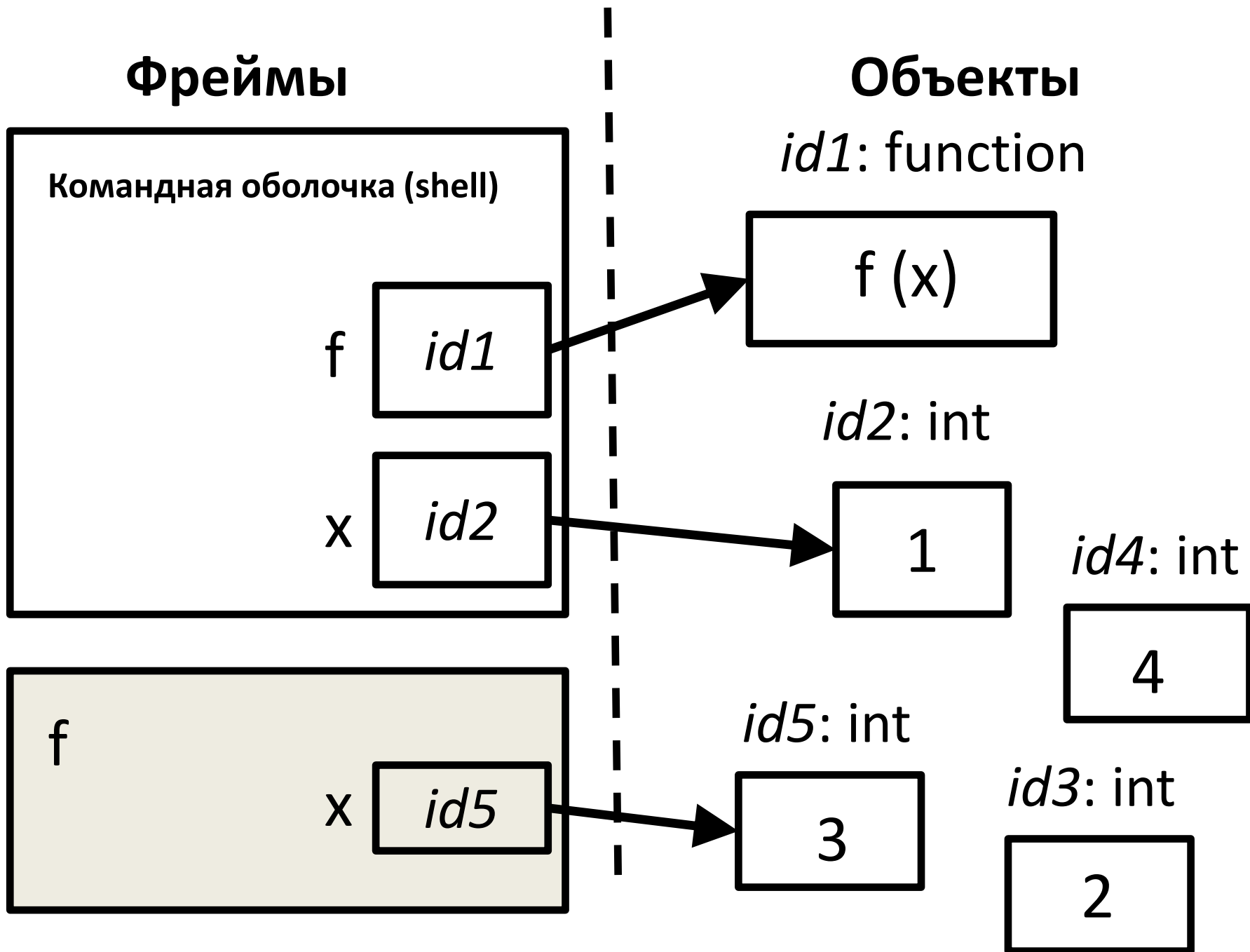
4

id5: int

3

id3: int

2



Пошаговое выполнение (трассировка)

```
>>> def f(x):
```

```
...     x = 2 * x
```

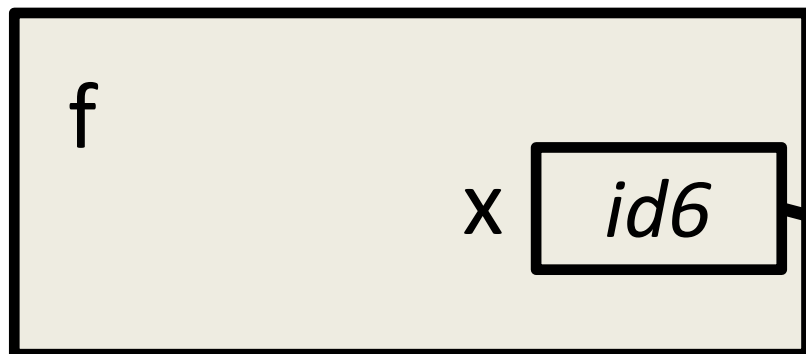
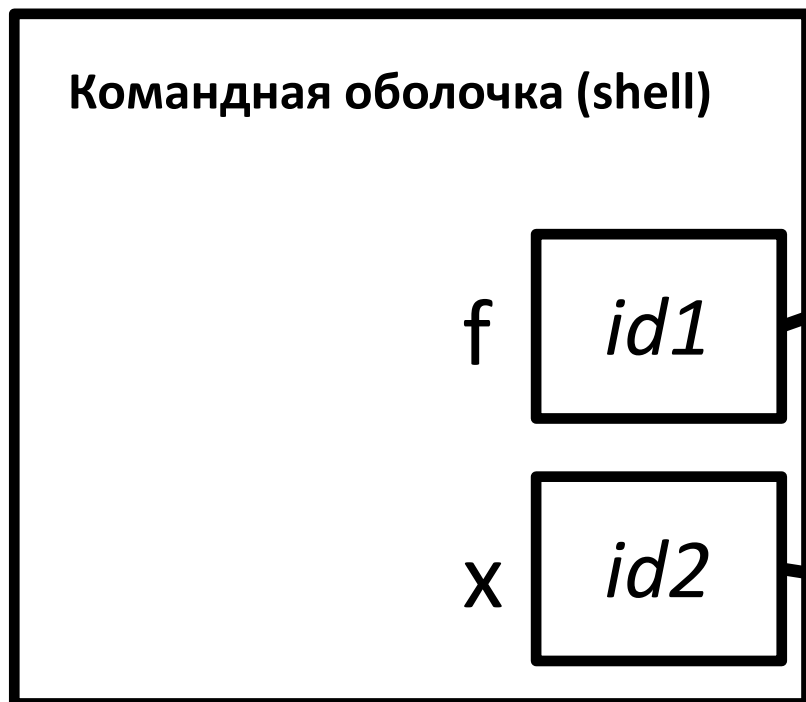
```
...     return x
```

```
>>> x = 1
```

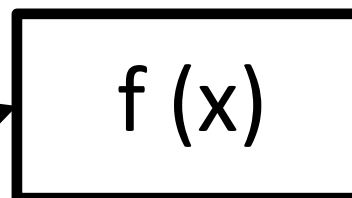
```
>>> x = f(x + 1) + f(x+2)
```

Фреймы

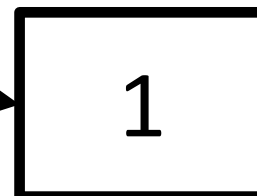
Объекты



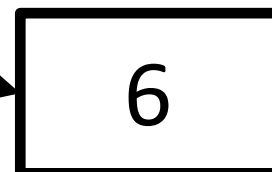
id1: function



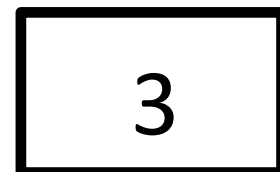
id2: int



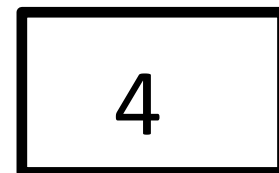
id6: int



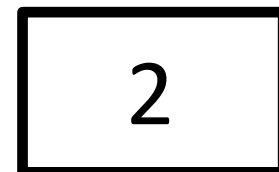
id5: int



id4: int



id3: int



Пошаговое выполнение (трассировка)

```
>>> def f(x):
```

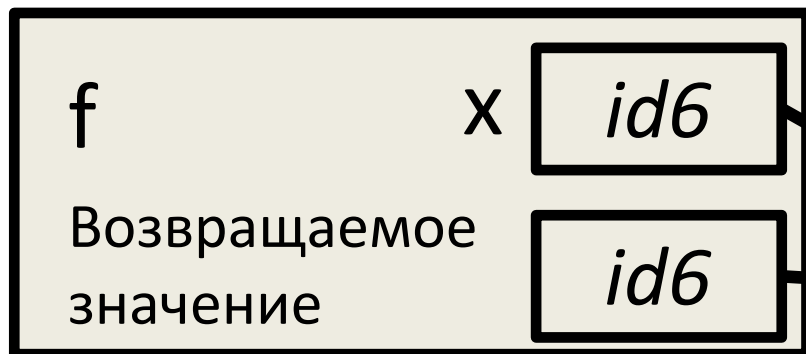
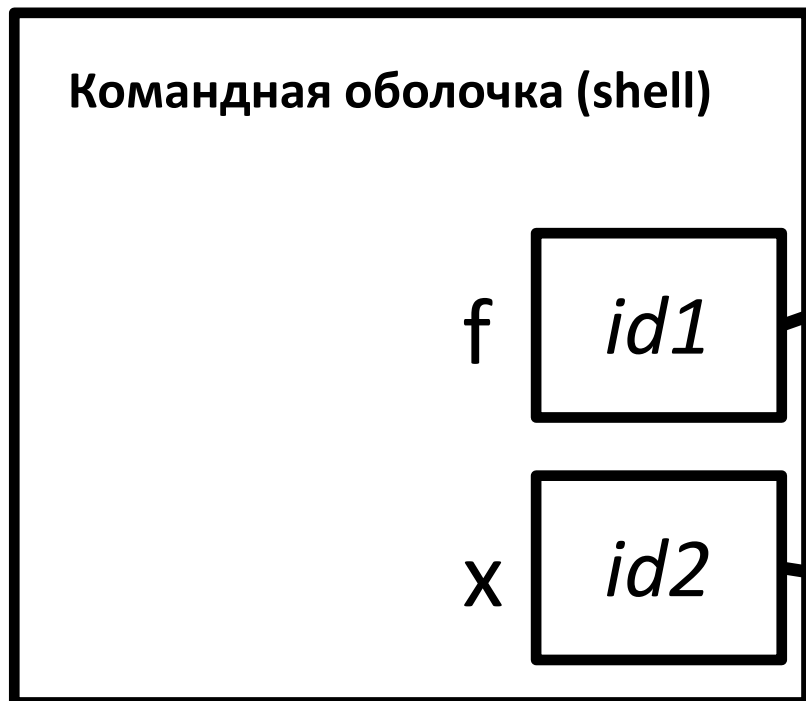
```
...     x = 2 * x
```

```
...     return x
```

```
>>> x = 1
```

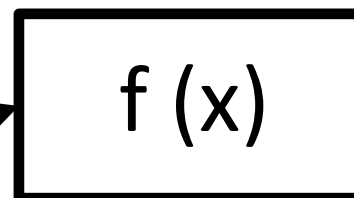
```
>>> x = f(x + 1) + f(x+2)
```


Фреймы

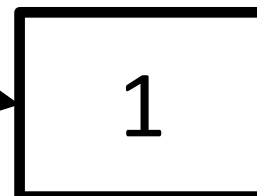


Объекты

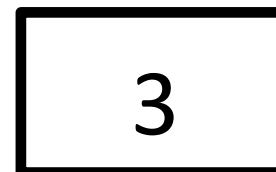
id1: function



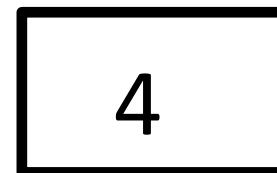
id2: int



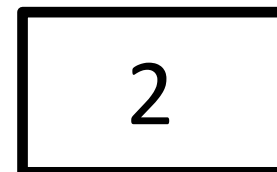
id5: int



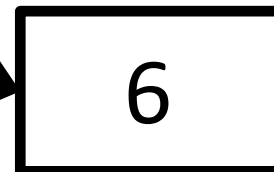
id4: int



id3: int



id6: int




Пошаговое выполнение (трассировка)

```
>>> def f(x):
```

```
...     x = 2 * x
```

```
...     return x
```

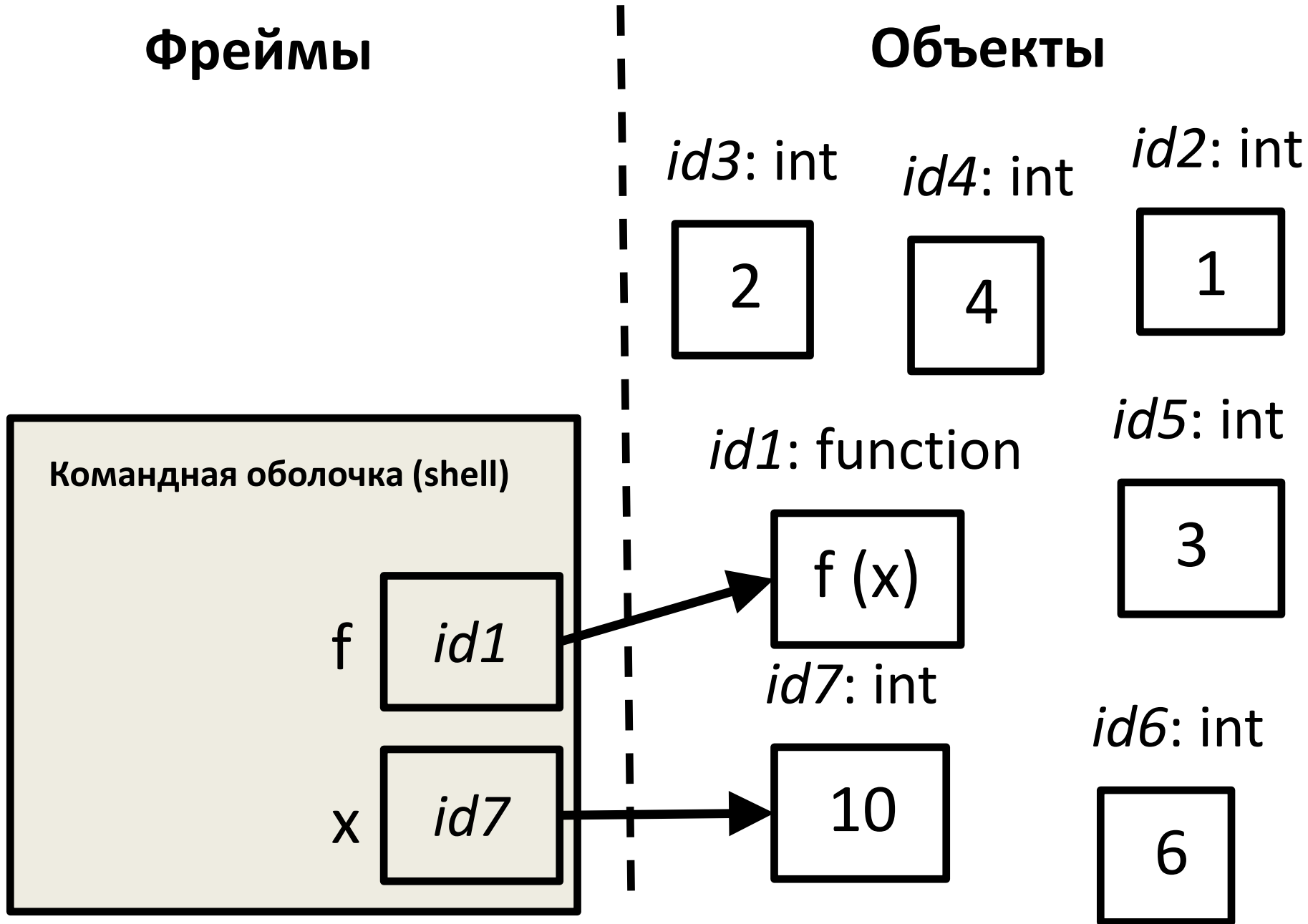
```
>>> x = 1
```



```
>>> x = f(x + 1) + f(x+2)
```

Фреймы

Объекты



Вопросы?

info@dfedorov.spb.ru

Исходные тексты: <http://dfedorov.spb.ru/python3/>