# Модуль glob

Модуль glob находит все пути, совпадающие с заданным шаблоном в соответствии с правилами, используемыми оболочкой Unix. Обрабатываются символы "\*" (произвольное количество символов), "?" (один символ), и диапазоны символов с помощью []. Для использования тильды "~" и переменных окружения необходимо использовать [os.path](https://pythonworld.ru/moduli/modul-os-path.html" \t "_blank).expanduser() и os.path.expandvars().

Для поиска спецсимволов, заключайте их в квадратные скобки. Например, [?] соответствует символу "?".

**glob.glob**(pathname) возвращение список (возможно, пустой) путей, соответствующих шаблону pathname. Путь может быть как абсолютным (например, /usr/src/Python-1.5/Makefile) или относительный (как ../../Tools/\*/\*.gif).

**glob.iglob**(pathname) - возвращает итератор, дающий те же значения, что и glob.glob.

**glob.escape**(pathname) - экранирует все специальные символы для glob ("?", "\*" и "["). Специальные символы в имени диска не экранируются (так как они там не учитываются), то есть в Windows escape("//?/c:/Quo vadis?.txt") возвращает "//?/c:/Quo vadis[?].txt". (новое в [python 3.4](https://pythonworld.ru/novosti-mira-python/python-340.html" \t "_blank)).

Рассмотрим, например, каталог, содержащий только следующие файлы: 1.gif, 2.txt и card.gif. glob.glob() вернёт следующие результаты. Обратите внимание, что любые ведущие компоненты пути сохраняются.

>>>

**>>> import** **glob**

**>>>** glob.glob('./[0-9].\*')

['./1.gif', './2.txt']

**>>>** glob.glob('\*.gif')

['1.gif', 'card.gif']

**>>>** glob.glob('?.gif')

['1.gif']

Если каталог содержит файлы, начинающиеся с ".", они не будут включаться по умолчанию. Рассмотрим, например, каталог, содержащий card.gif и .card.gif:

>>>

**>>> import** **glob**

**>>>** glob.glob('\*.gif')

['card.gif']

**>>>** glob.glob('.c\*')

['.card.gif']