### re.match(pattern, string)

Этот метод ищет по заданному шаблону в начале строки. Например, если мы вызовем метод match() на строке «AV Analytics AV» с шаблоном «AV», то он завершится успешно. Но если мы будем искать «Analytics», то результат будет отрицательный:

### re.search(pattern, string)

Метод похож на match(), но ищет не только в начале строки. В отличие от предыдущего, search() вернёт объект, если мы попытаемся найти «Analytics»:

result = re.search(r'Analytics', 'AV Analytics Vidhya AV')

print result.group(0)

Результат:

Analytics

Метод search() ищет по всей строке, но возвращает только первое найденное совпадение.

### re.findall(pattern, string)

Возвращает список всех найденных совпадений. У метода findall() нет ограничений на поиск в начале или конце строки. Если мы будем искать «AV» в нашей строке, он вернет все вхождения «AV». Для поиска рекомендуется использовать именно findall(), так как он может работать и как re.search(), и как re.match().

### re.split(pattern, string, [maxsplit=0])

Этот метод разделяет строку по заданному шаблону.

### re.sub(pattern, repl, string)

Ищет шаблон в строке и заменяет его на указанную подстроку. Если шаблон не найден, строка остается неизменной.

### re.compile(pattern, repl, string)

Мы можем собрать регулярное выражение в отдельный объект, который может быть использован для поиска. Это также избавляет от переписывания одного и того же выражения.