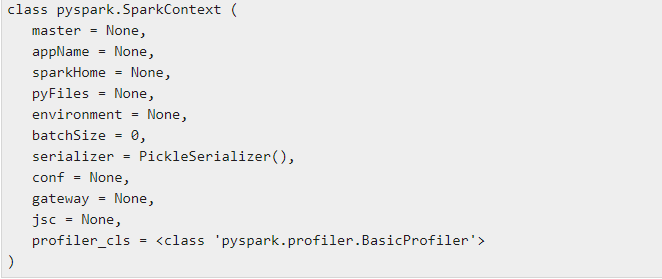
**Spark Property**

Thuộc tính Spark sẽ là nơi kiểm soát, cài đặt ứng dụng của chúng ta. Các thuộc tính này được cài đặt thông qua SparkContext. Và khi cài đặt bất kì chương trình nào, thì điều đầu tiên chúng ta làm sẽ làm cài đặt SparkContext và ứng dụng sẽ dựa trên các property đã set trong SparkContext để chạy ứng dụng.

Trong thư viện PySpark thì đã cài đặt sẵn SparkContext, việc của bạn là cài đặt cho các property của nó.



Ảnh các property trong SparkContext.

Các propery thường dùng trong SparkContext:

* Master: Đây là URL ứng dụng sẽ kết nối tới.
* appName: tên ứng dụng của bạn.

Phải import thư viện vô trước khi dùng



Sau đó set master kết nối tới “local” còn tên ứng dụng là “word counting”.



Hoặc



Và mỗi thời điểm ta chỉ có thể tạo một context, nếu muốn hủy thì phải gọi stop.



**Spark RDD**

RDD (Resilent Distributed Dataset) đây là các phần tử giúp việc xử lý dữ liệu được chia nhỏ và xử lý tác vụ một cách song song. RDD khi chúng ta đã khởi tạo thì chúng ta không thể thay đổi được nó, đồng thời nó cũng có khả năng chịu lỗi, nghĩa là khi có lỗi thì nó sẽ tự khắc phụ lỗi đó.

Có 2 hoạt động trên RDD:

* Chuyển đổi: đây là dùng để thực thi các hàm như filter, group by, map, …
* Thực thi: Đây là nơi thực hiện các hàm tính toán để ra kết quả.

Để truyền dữ liệu một cách song song thì ta sẽ dùng hàm parallelize. Các phần tử được truyền vào sẽ được phân tán để có thể vận hành một cách song song.



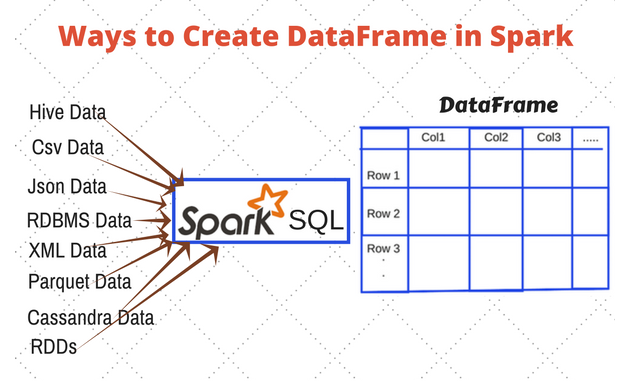
Đoạn trên sẽ đẩy một mảng text để thực thi chương trình mà ta muốn.

Các hàm thường được sử dụng trong Spark RDD:

* Count(): đếm các phần tử trong RDD
* Collect(): trả các phần tử trong RDD
* Filter(f): trả ra RDD mới sau khi lọc điều kiện f với từng phần tử trong RDD.
* Map(f): trả ra RDD mới sau khi map dữ liệu mới trong f với từng phần tử trong RDD.
* Reduce(f): dùng để trả ra kết quả cuối cùng sau khi kết hợp giữa các phần tử trong RDD qua thuật toán f.
* reduceByKey(f): dùng để trả ra kết quả cuối cùng sau khi kết hợp giữa các phần tử thông qua key trong RDD và thông qua thuật toán f.

**Spark DataFrame**

DataFrame là dạng dữ liệu theo cột, hàng giống như một dạng Excel. Để tạo một DataFrame thì ta chỉ cần đọc data từ JSON, CSV, tải dữ liệu từ RDD có sẵn. Nó giúp ta thao tác dữ liệu như một bảng dữ liệu rất nhiều hàng và cột.



Đọc file, cụ thể là file csv:



Map data trong df để ra dạng dữ liệu mà ta muốn



Kết quả:

