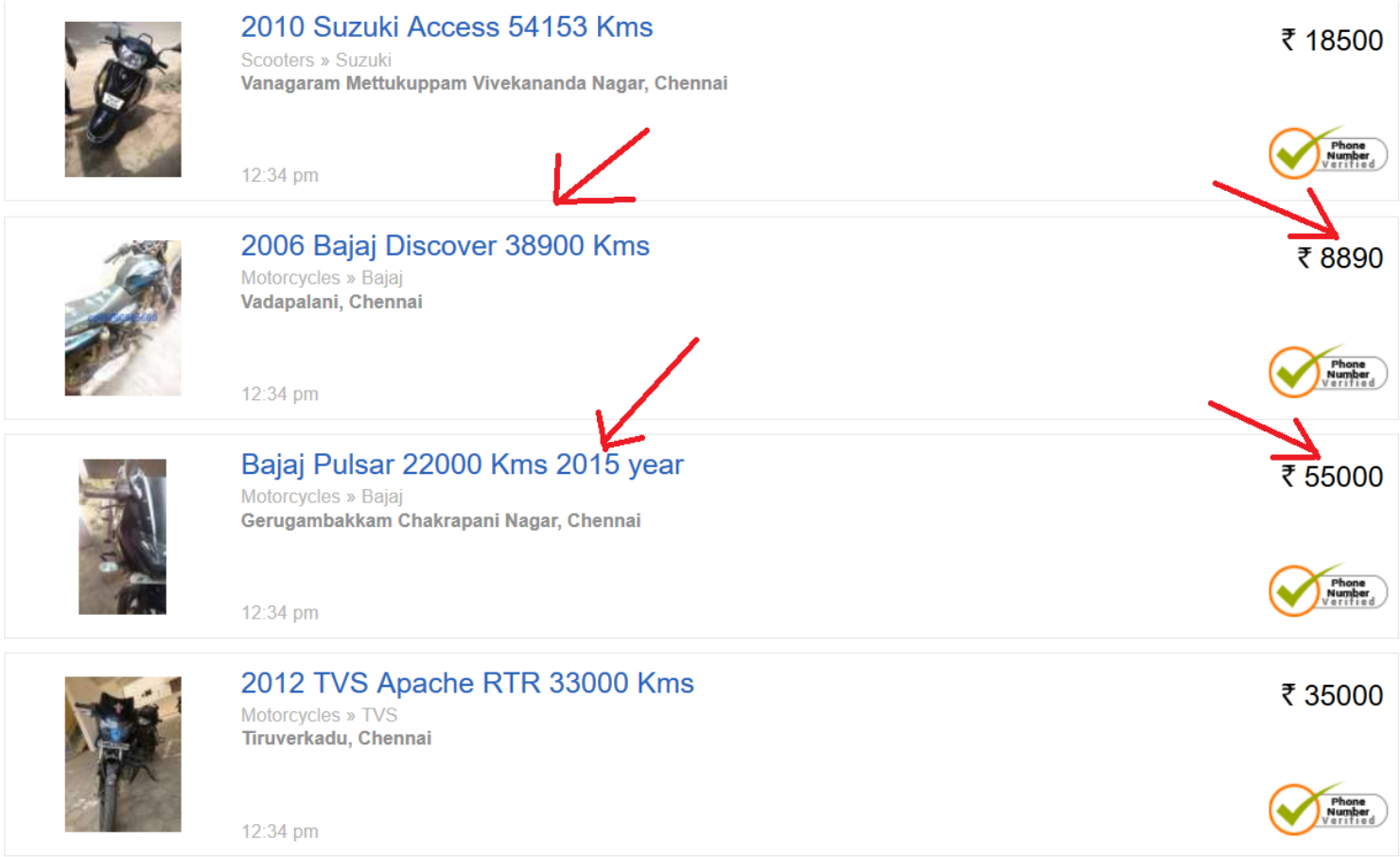
Getting started with scrapy

Scraping olx.com for bikes and cars

Web-scraping

Extracting information from web pages and following links between pages



Scrapy

- Python framework for web scraping (and web api extraction)
- Built on top of the **Twisted** asynchronous networking library
- Extensible

Code: Scrapy spider

- Class inheriting from scrapy.Spider
- Must have a name attribute which should be unique in the project. This is used to identify the spider
- start_urls specifies a list of urls to crawl additional links may be followed and crawled
- parse(self, response) function with the logic to handle the response received should yield scrapy.Item or scrapy.Request or dict
- Scrapy.Request objects are used follow links. It must specify url and callback function to parse the response

Code example 1: single file

- Self contained file. Simple.
- Suitable for one time extraction / small works

Code example 2: project structure

- Used to group multiple spiders in the same project.
- Allows for additional settings
- Handling the results

Code example 3: Item and ItemLoader

- scrapy.Item allows for creating schema crawled data.
- Allows additional rules/sanity checks and preprocessing for ach field.
- Separtion for extraction code and data processing code.

- scrapy.loader.ItemLoader convenient mechanism for populating Items and parsing.
- Allows custom pre-processor definitions for pre-processing

Questions

This was a very small introduction to scrapy.

- https://github.com/shanmuga-cv/scrapy-getting-started

For detailed tutorial refer

- https://doc.scrapy.org/en/latest/intro/tutorial.html
- https://www.youtube.com/watch?v=vkA1cWN4DEc&list=PLZyvi_9gamL-EE3zQJbU5N3nzJcfNeFHU

About me

BigData, Python, Macine-Learning

Scraping???

https://github.com/shanmuga-cv/

https://twitter.com/sachi_vel

Shanmuga.chidambaravel@gmail.com