**Assignment 4- Spider for Amazon.in webpage**

1. **Create a Web Scraper for following web page.**

<https://www.amazon.in/s?k=python+programming+books&crid=2T7NO081U4V9P&sprefix=python%2Caps%2C230&ref=nb_sb_ss_ts-doa-p_1_6>

1. **Scrap Book Name and Price and Write them in following MySQL/MongoDB Table/Document.**

|  |  |
| --- | --- |
| **Book** | **Price** |
|  |  |
|  |  |

import scrapy  
from ..items import QuotescrapItem  
  
  
class QuoteSpider(scrapy.Spider):  
 name = 'quotes' # spider name  
 start\_urls = ['https://www.amazon.in/s?k=python+programming+books&crid=2T7NO081U4V9P&sprefix=python%2Caps%2C230&ref=nb\_sb\_ss\_ts-doa-p\_1\_6'] # url to scrape  
  
 def parse(self,response):  
  
 items = QuotescrapItem()  
 all\_div\_quotes = response.css('.puisg-row')  
 for quotes in all\_div\_quotes:  
 book\_name = quotes.css('.a-text-normal::text').extract()[0]  
 price = quotes.css('.a-price-whole::text').extract()[0]  
 items['book\_name'] = book\_name  
 items['price'] = price  
 if book\_name != ' ' and price != ' ':  
 yield items

import scrapy  
  
  
class QuotescrapItem(scrapy.Item):  
 # define the fields for your item here like:  
 book\_name = scrapy.Field()  
 price = scrapy.Field()

import pymongo  
  
  
class QuotescrapPipeline:  
  
 def \_\_init\_\_(self):  
 self.conn = pymongo.MongoClient('mongodb://localhost:27017')  
 db = self.conn['amz\_books']  
 self.collection = db['amz\_books\_db']  
  
 def process\_item(self, item, spider):  
 self.collection.insert\_one(dict(item))  
 return item