\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

---PDF to Elastic search runbook---

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Step 1 : Download Elastic Search Zip/Pkg from below link

https://www.elastic.co/downloads/elasticsearch

Step 2 : Extract the above file

Step 3 : Open terminal, move to extracted dir

cd ~elasticsearch-7.3.0/bin/

Step 4 : Run elasticsearch.bat

$~elasticsearch-7.3.0/bin> elasticsearch.bat

Step 5: Check elatic search from browser,

localhost:9200

you ll get response like below

*{*

*"name" : "EQ-EQ6283311",*

*"cluster\_name" : "elasticsearch",*

*"cluster\_uuid" : "AvNuOAqpRlmnoGOdlXvTOw",*

*"version" : {*

*"number" : "7.3.0",*

*"build\_flavor" : "default",*

*"build\_type" : "zip",*

*"build\_hash" : "de777fa",*

*"build\_date" : "2019-07-24T18:30:11.767338Z",*

*"build\_snapshot" : false,*

*"lucene\_version" : "8.1.0",*

*"minimum\_wire\_compatibility\_version" : "6.8.0",*

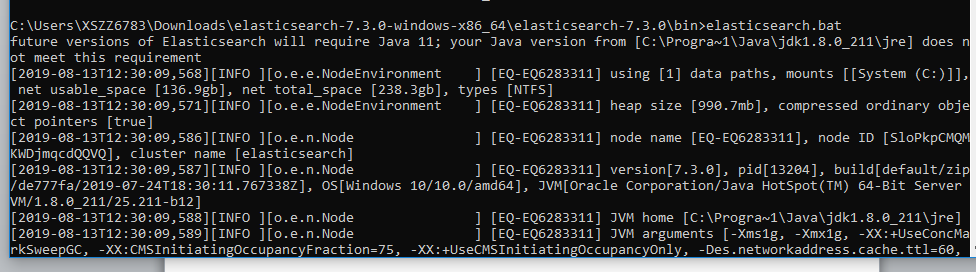
*"minimum\_index\_compatibility\_version" : "6.0.0-beta1"*

*},*

*"tagline" : "You Know, for Search"*

*}*

Its up...



Step 6 : Download Kibana

<https://www.elastic.co/downloads/kibana>

Step 7 : Extract the above file

Step 8 : Open terminal, move to extracted dir

cd ~ kibana-7.3.0-windows-x86\_64/bin/

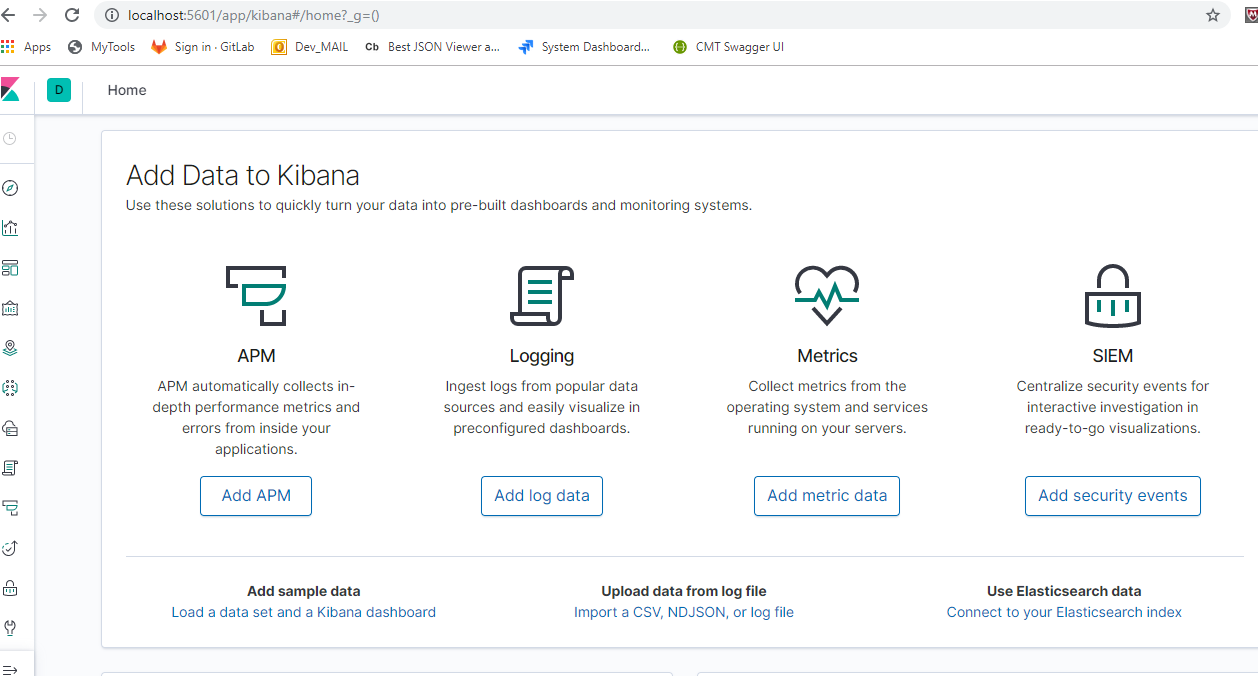
Step 9 : Run Kibana.bat

$kibana-7.3.0-windows-x86\_64\bin

Step 10 : Check Kibana

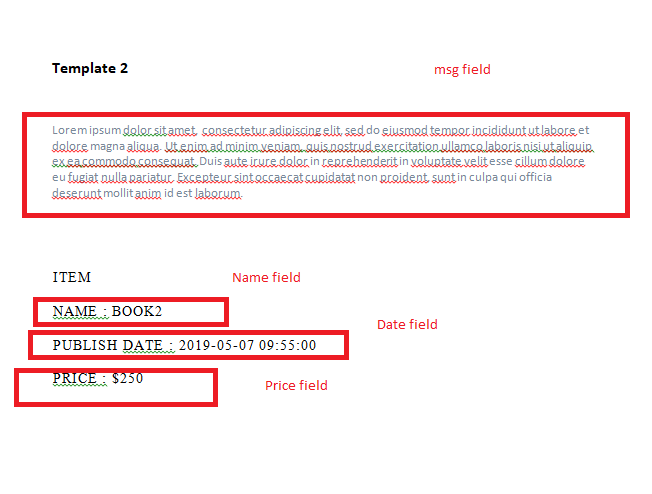
<http://localhost:5601/>

You ll get below page



Below is the sample pdf

* Four columns has been extracted and inserted into Elastic search



Assuming Python is setup already

Step 11 : Use below python code to extract Pdf

import PyPDF2

import re

import requests

import json

from datetime import date

class ElasticModel:

name = ""

price = ""

msg = ""

date = ""

def toJSON(self):

return json.dumps(self, default=lambda o: o.\_\_dict\_\_,

sort\_keys=True, indent=4)

# pdf file object

# you can find find the pdf file with complete code in below

pdfFileObj = open('Your PDF Path', 'rb')

# pdf reader object

pdfReader = PyPDF2.PdfFileReader(pdfFileObj)

# number of pages in pdf

print(pdfReader.numPages)

# a page object

pageObj = pdfReader.getPage(0)

# extracting text from page.

# this will print the text you can also save that into String

line = pageObj.extractText()

line = line.replace("\n","")

print(line)

matchObj = re.match( r'NAME', line, re.M|re.I)

NET\_FIELDS = 4

m = [re.search('NAME : (.+?)PUBLISH', line), #Regex Name extract

re.search('PUBLISH DATE : (.+?)PRICE', line), #Regex extract Publish Date

re.search('PRICE : (.\*)', line), #Regex extract Price

re.search('Template 1 (.+?)ITEM', line)] #Regex extract msg

i=0

print("==> Net fields : " + str(NET\_FIELDS))

eModel = ElasticModel();

while i < 4 :

if m[i]:

# while matchObj is not None:

print("matchObj.group() : "+ m[i].group(1))

if i == 0 :

eModel.name = m[i].group(1).strip()

elif i == 1:

eModel.date = m[i].group(1).strip()

elif i == 2:

eModel.price = m[i].group(1).strip()

else:

eModel.msg = m[i].group(1).strip()

else:

print("No match!!")

i = i+1

def \_\_sendToElasticSearch\_\_(elasticModel):

print("Name : " + str(eModel))

url = "http://localhost:9200/mypdfcollectiontest/\_doc?pretty"

data = elasticModel.toJSON()

#data = serialize(eModel)

response = requests.post(url, data=data,headers={

'Content-Type':'application/json',

'Accept-Language':'en'

})

print("Url : " + url)

print("Data : " + str(data))

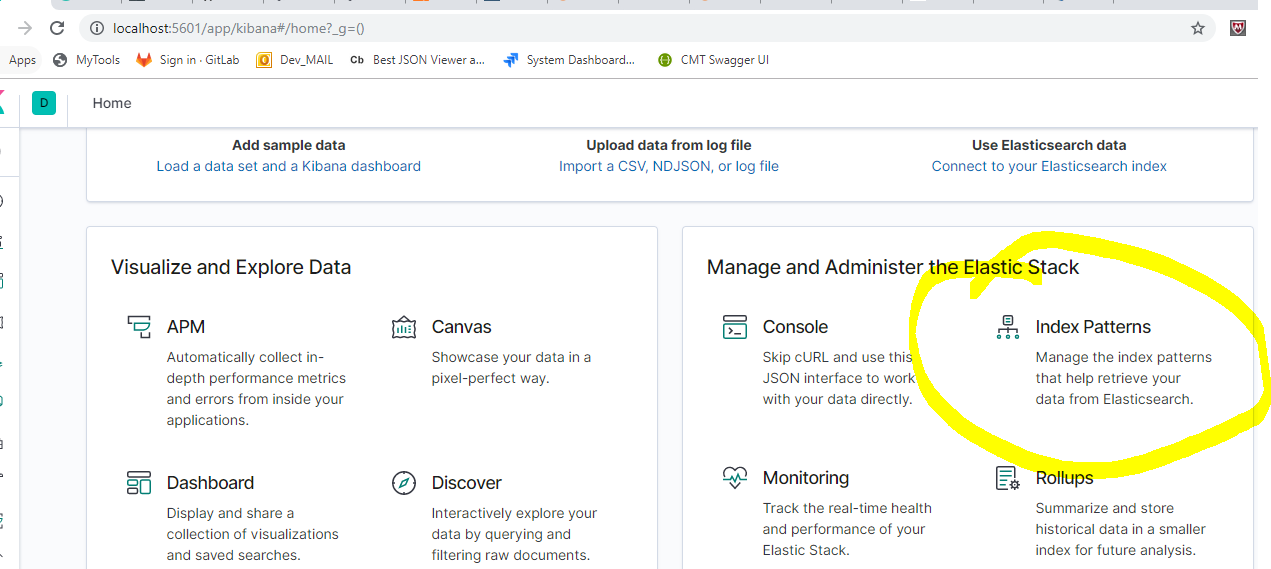
print("Request : " + str(requests))

print("Response : " + str(response))

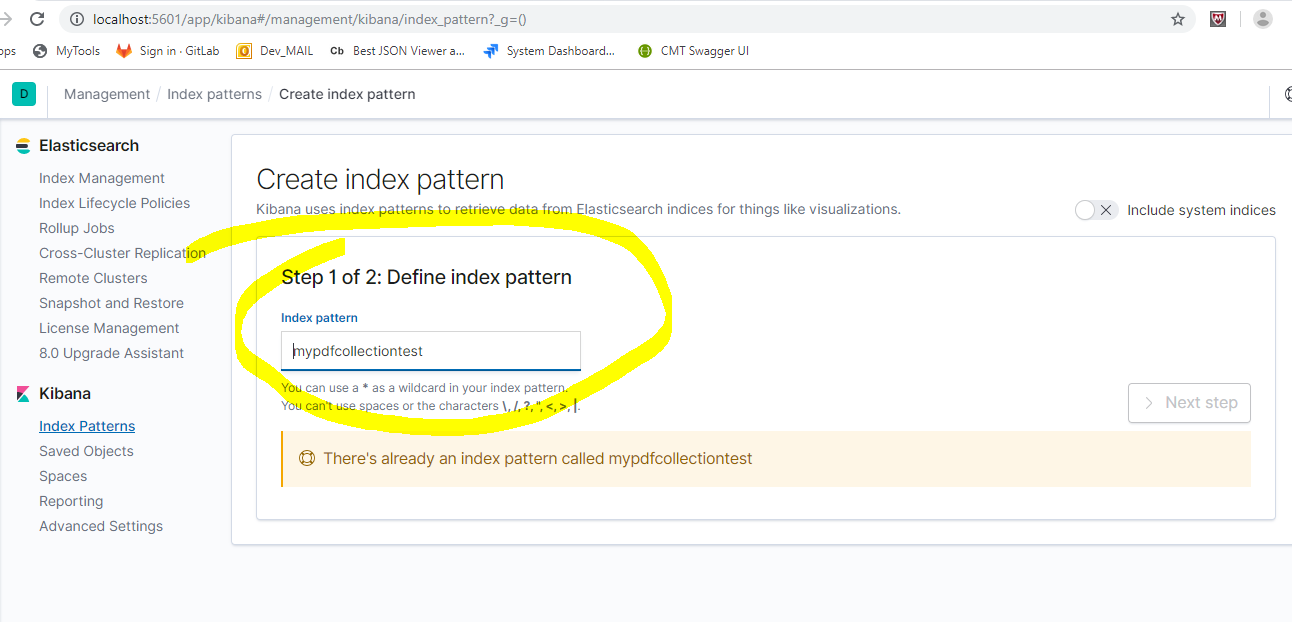
\_\_sendToElasticSearch\_\_(eModel)

Step 12 : Change the Pdf Path in above code and Run the python script in Jupiter

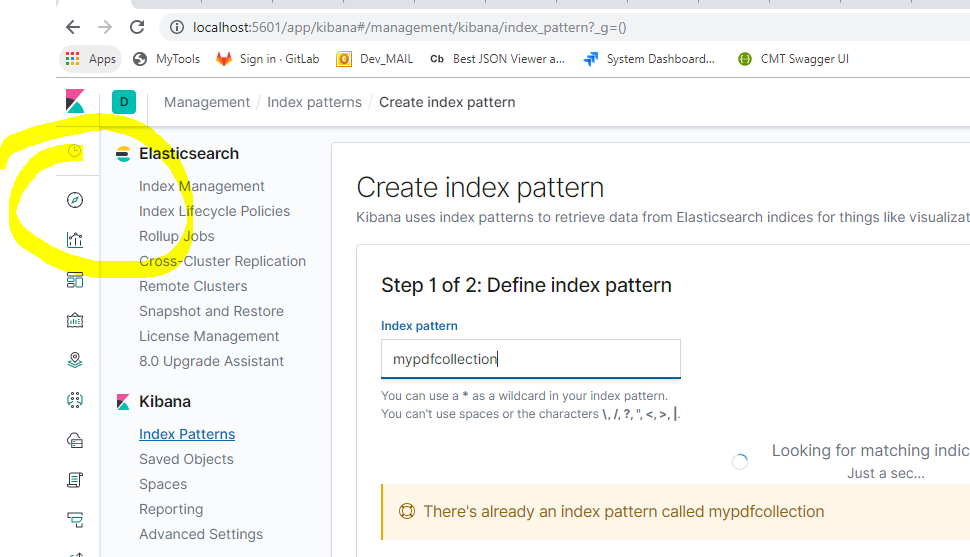
Step 13 : Go To Kibana Home page, click on index patterns



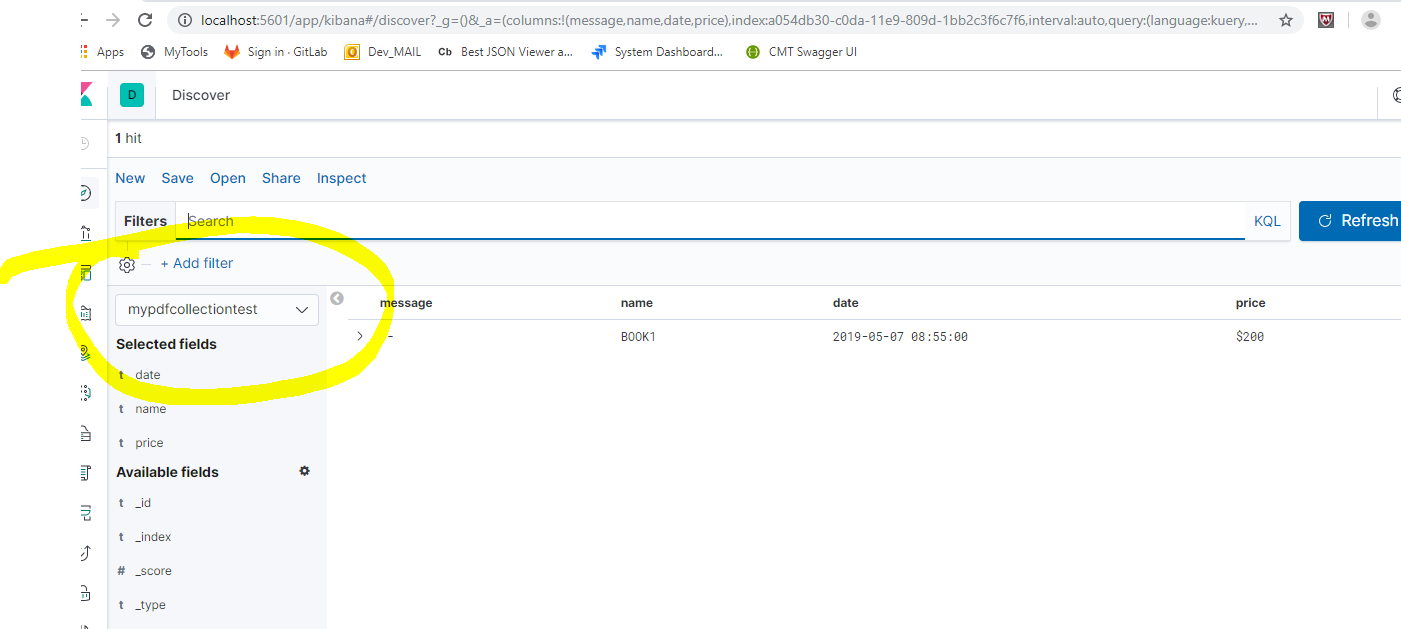
Step 14: Click on Create index, and add index pattern “mypdfcollectiontest"



Step 15: Click next step and finish, then click on Discover button



Step 16: Select mypdfcollectiontest index to see inserted pdf doc



Step 17 : From the available fields add Extracted fields like Name, Price etc

