**WEB MINING**

**HARSH TRIPATHI**

**16BCE1186**

**1**. **Given a root URL, e.g., "Vit.ac.in", Design a simple crawler to return all the pages (URL) from this site.**

**2. Given a root URL, e.g., "Vit.ac.in", Design a simple crawler to return all the pages (URL) with depth level =2 from this site.**

**PROGRAM**

from BeautifulSoup import BeautifulSoup

import urllib2

import re

html\_page = urllib2.urlopen("http://www.vit.ac.in")

soup = BeautifulSoup(html\_page)

links = []

for link in soup.findAll('a', attrs={'href': re.compile("^http://")}):

links.append(link.get('href'))

link = set(links)

for i in links:

html\_page = urllib2.urlopen(i)

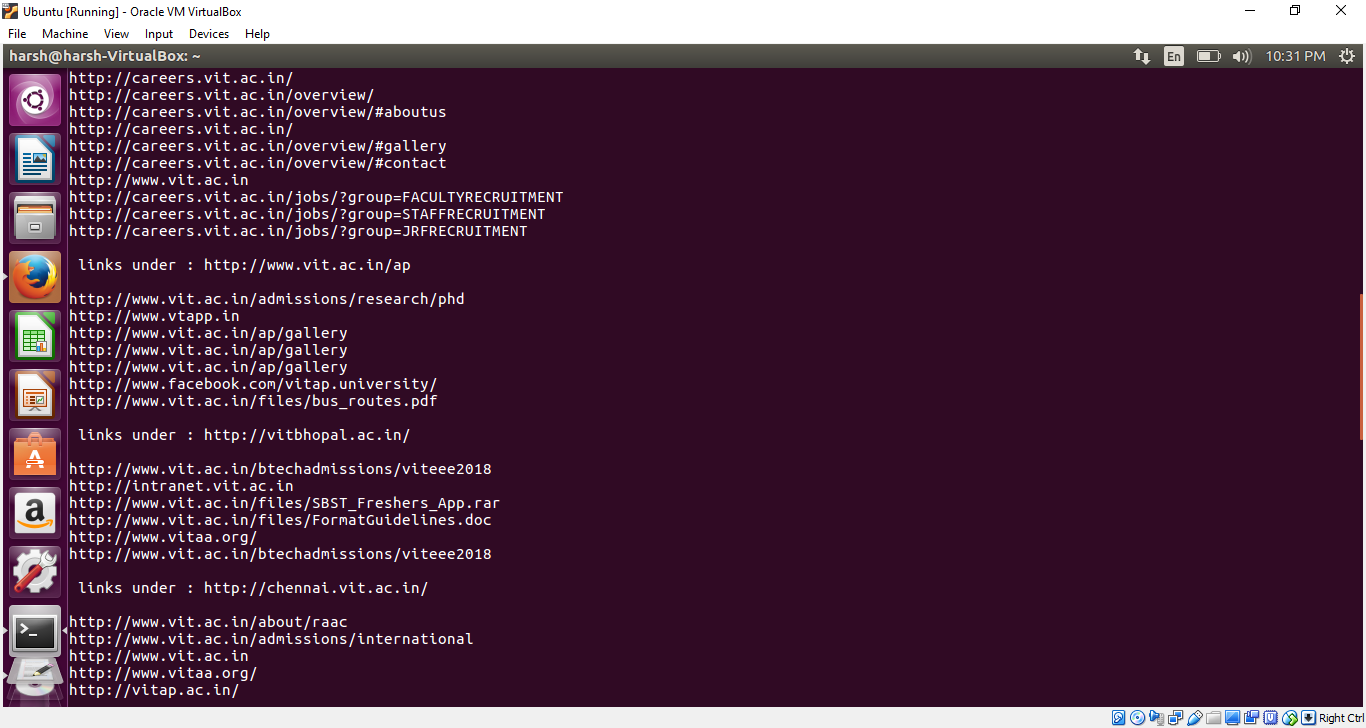
soup = BeautifulSoup(html\_page)

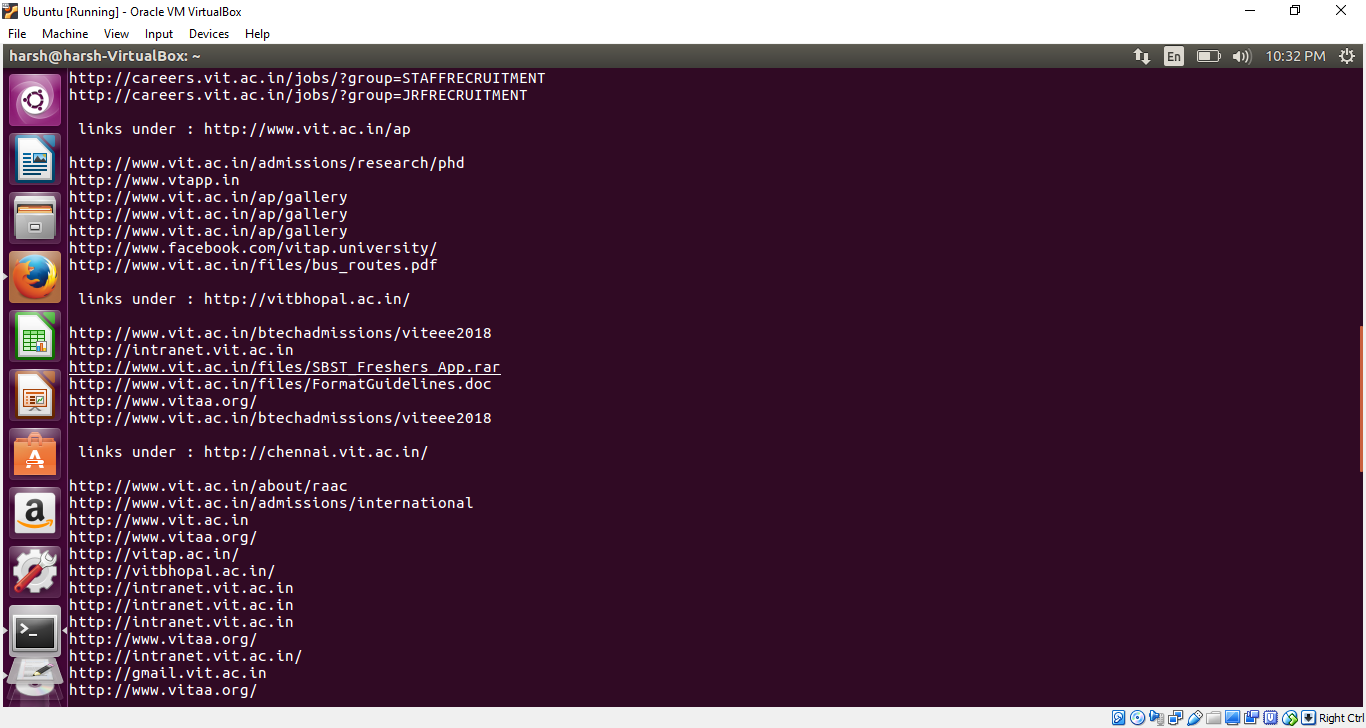
print('\n links under : '+ i + '\n')

for link in soup.findAll('a', attrs={'href': re.compile("^http://")}):

print(link.get('href'))

**OUTPUT**

****

****

1. **Given a root URL, e.g., "Vit.ac.in", Design a simple crawler to return all URL of the pages with the word “<b> student </b>”from this site.**

**PROGRAM**

from BeautifulSoup import BeautifulSoup

import urllib2

import re

html\_page = urllib2.urlopen("https://www.vit.ac.in /")

soup = BeautifulSoup(html\_page)

links = []

for link in soup.findAll('a', attrs={'href': re.compile("^http://")}):

links.append(link.get('href'))

std = soup.findAll('b', text = re.compile('student'))

std = soup.findAll('b', text = re.compile('student')) #regex to match student and vit in level1

print(std)

link = set(links) #to remove duplicate links

for i in links:

html\_page = urllib2.urlopen(i)

soup = BeautifulSoup(html\_page)

for link in soup.findAll('a', attrs={'href': re.compile("^http://")}):

html\_page = urllib2.urlopen(link)

soup = BeautifulSoup(html\_page)

std = soup.findAll('b', text = re.compile('student'))

vit = soup.findAll('b', text = re.compile('vit')) #regex to match student and vit in level2

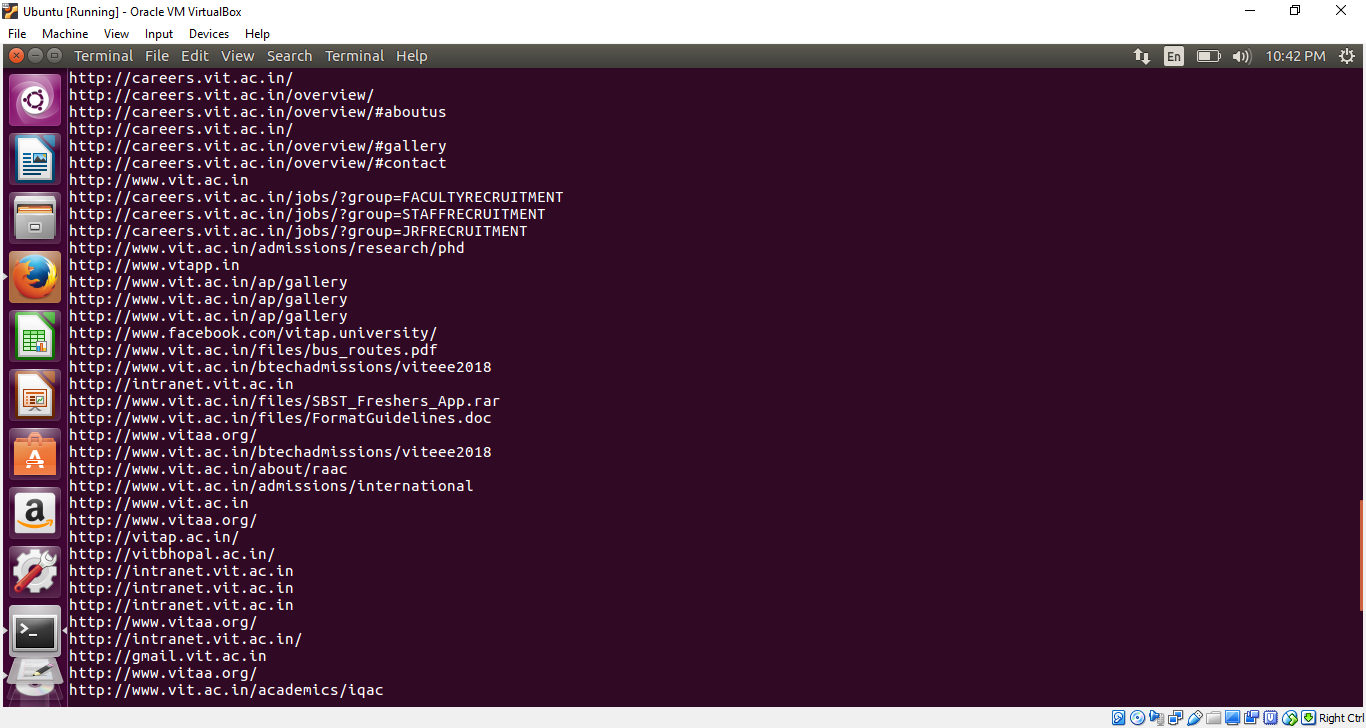
print(std)

print(vit)

**OUTPUT**

**ALL THESE LINKS CONTAIN EITHER WORD STUDENT OR VIT IN BOLD TAG.**

**CRAWLED UPTO DEPTH 2**

****