1. Which of the following extracts information from user generated content?

Ans:-Web Scraping

2. Which of the following is not a web scraping library in python?

Ans:-Scrapy

3 .Selenium tests \_\_\_\_\_\_\_\_\_\_?

Ans:- Browser based applications

4. Task of crawling is performed by a complex software which is known as:

Ans:- Crawler

5. Which of the following commands is used to access name of a tag in Beautiful Soup?

Ans:- tag.name

6. Which of the following is the default parser in Beautiful Soup?

Ans:-xlml

7. In selenium the webdriver is used to?

Ans:-to download any content from a webpage

8. In selenium, driver**.find\_elements\_by\_xpath(‘given xpath’)** returns:

Ans: the list of all webelements associated with the ‘given xpath’

9. The script **‘window.scrollBy(0,a)** scrolls the webpage by?

Ans:- **‘a’** number of pixels vertically

10. Which of the following is(are) tags of HTML?

Ans:- <a> , <href>

11. What is the main difference between a web scraper and a web crawler?

Ans: Unlike web crawling, a web scraper searches for specific information on specific websites or pages. Basically, web crawling creates a copy of what's there and web scraping extracts specific data for analysis, or to create something new. ... Web crawling would be generally what Google, Yahoo, Bing etc.

12. What is **‘robots.txt’** file? What is the use of **‘robots.txt’** file?

Ans:- A robots. txt file is a simple text file which webmasters can create to tell web crawlers which parts of a website should be crawled and which should not. The file is stored in the main directory (root) on the server. When a crawler arrives at a website, it first reads the robots.

13. What are static and dynamic web pages?

Ans: Static Web pages are very simple. ... In static web pages, Pages will remain the same until someone changes it manually. Dynamic Web Pages: Dynamic Web Pages are written in languages such as CGI, AJAX, ASP, ASP.NET, etc. In dynamic web pages, the Content of pages is different for different visitors.

14. Write a python program to check whether a webpage contains a title or not.

**Ans:-** from urllib.request import urlopen

from urllib.error import HTTPError

from bs4 import BeautifulSoup

def getTitle(url):

try:

html = urlopen(url)

except HTTPError as e:

return None

try:

bsObj = BeautifulSoup(html.read(), "lxml")

title = bsObj.body.h1

except AttributeError as e:

return None

return title

title = getTitle(url)

if title == None:

return "Title could not be found"

else:

return title

print(getTitle("https://www.w3resource.com/"))

print(getTitle("http://www.example.com/"))