{

"cells": [

{

"cell\_type": "code",

"execution\_count": null,

"id": "c333125b",

"metadata": {},

"outputs": [],

"source": [

"#Q1 \n",

"#Importing required libraries\n",

"import selenium\n",

"import pandas as pd\n",

"from selenium import webdriver\n",

"import time\n",

"from selenium.common.exceptions import NoSuchElementException\n",

"from selenium.webdriver.common.keys import Keys\n",

"from selenium.webdriver.common.by import By\n",

"import warnings\n",

"warnings.filterwarnings(\"ignore\")\n"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "b434ebb9",

"metadata": {},

"outputs": [],

"source": [

"driver = webdriver.Chrome(\"chromedriver.exe\")\n",

"time.sleep(2)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "8aa4ea29",

"metadata": {},

"outputs": [],

"source": [

"driver.get('https://en.wikipedia.org/wiki/List\_of\_most-viewed\_YouTube\_videos')"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "8cfc6c15",

"metadata": {},

"outputs": [],

"source": [

"rank =[]\n",

"name=[]\n",

"artist=[]\n",

"upload\_date=[] \n",

"views=[]"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "9955b295",

"metadata": {},

"outputs": [],

"source": [

"n=driver.find\_elements(By.XPATH,\"//table[@class='wikitable sortable jquery-tablesorter']/tbody/tr/td[2]\")\n",

"for i in n:\n",

" name.append(i.text) \n",

"print(name[0:30])"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "da466750",

"metadata": {},

"outputs": [],

"source": [

"r=driver.find\_elements(By.XPATH,\"//table[@class='wikitable sortable jquery-tablesorter']/tbody/tr/td[1]\") \n",

"for i in r: \n",

" rank.append(i.text)\n",

"print(rank[0:30]) "

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "c80ffcff",

"metadata": {},

"outputs": [],

"source": [

"artst=driver.find\_elements(By.XPATH,\"//table[@class='wikitable sortable jquery-tablesorter']/tbody/tr/td[3]\")\n",

"for i in artst: \n",

" artist.append(i.text)\n",

"print(artist[0:30]) \n"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "bae20721",

"metadata": {},

"outputs": [],

"source": [

"vws=driver.find\_elements(By.XPATH,\"//table[@class='wikitable sortable jquery-tablesorter']/tbody/tr/td[4]\")\n",

"for i in vws:\n",

" views.append(i.text)\n",

"print(views[0:30]) "

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "dac66812",

"metadata": {},

"outputs": [],

"source": [

"ud=driver.find\_elements(By.XPATH,\"//table[@class='wikitable sortable jquery-tablesorter']/tbody/tr/td[5]\")\n",

"for i in ud: \n",

" upload\_date.append(i.text)\n",

"print(upload\_date[0:30]) \n",

" "

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "a52c28a1",

"metadata": {},

"outputs": [],

"source": [

"#Checking the length of data extracted\n",

"print(len(name),len(rank),len(upload\_date),len(views),len(artist))"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "ad4f42bd",

"metadata": {},

"outputs": [],

"source": [

"Youtube=pd.DataFrame({})\n",

"Youtube['Rank']=rank\n",

"Youtube['Name']=name\n",

"Youtube['Artist']=artist\n",

"Youtube['Views']=views\n",

"Youtube['Upload Date']=upload\_date\n",

"Youtube"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "cfca9571",

"metadata": {

"scrolled": true

},

"outputs": [],

"source": [

"#Removing the stray numbers from videoname\n",

"yt\_new=Youtube[\"Name\"].str.split(\"[\", n = 1, expand = True) \n",

"yt\_new"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "a9deeed3",

"metadata": {},

"outputs": [],

"source": [

"#Dropping the column with stray numbers\n",

"Youtube.drop(columns=['Name'],axis=1,inplace=True)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "384a093e",

"metadata": {},

"outputs": [],

"source": [

"#Inserting the name column\n",

"Youtube.insert(1,\"Name\",yt\_new[0])"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "ba9e56db",

"metadata": {},

"outputs": [],

"source": [

"#Checking the data after removing the stray numbers\n",

"Youtube "

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "6790c42c",

"metadata": {},

"outputs": [],

"source": [

"driver.close()"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "8bea5591",

"metadata": {},

"outputs": [],

"source": [

"#Q2...............\n",

"\n",

"import selenium\n",

"import pandas as pd\n",

"from selenium import webdriver\n",

"import time\n",

"import warnings\n",

"warnings.filterwarnings(\"ignore\")"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "873db452",

"metadata": {},

"outputs": [],

"source": [

"driver = webdriver.Chrome(\"chromedriver.exe\")\n",

"time.sleep(2)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "b78bb5e0",

"metadata": {},

"outputs": [],

"source": [

"driver.get('https://www.bcci.tv/')"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "365aa239",

"metadata": {},

"outputs": [],

"source": [

"link\_button=driver.find\_element\_by\_xpath('//\*[@id=\"navigation\"]/ul[1]/li[2]/a')\n",

"link\_button.click()"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "7460067c",

"metadata": {},

"outputs": [],

"source": [

"match\_title=[]\n",

"series=[]\n",

"place=[]\n",

"Date=[]\n",

"Time=[]"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "c104d637",

"metadata": {},

"outputs": [],

"source": [

"title=driver.find\_elements\_by\_xpath('//span[@class=\"matchOrderText ng-binding ng-scope\"]')\n",

"for i in title:\n",

" match\_title.append(i.text)\n",

"print(match\_title)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "08746216",

"metadata": {},

"outputs": [],

"source": [

"srs=driver.find\_elements\_by\_xpath('//span[@class=\"ng-binding\"]')\n",

"for i in srs:\n",

" series.append(i.text)\n",

"print(series)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "8fe13230",

"metadata": {},

"outputs": [],

"source": [

"loc=driver.find\_elements\_by\_xpath('//span[@class=\"ng-binding ng-scope\"]')\n",

"for i in loc:\n",

" place.append(i.text)\n",

"print(place)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "e7acfb9a",

"metadata": {},

"outputs": [],

"source": [

"date=driver.find\_elements\_by\_xpath('//h5[@class=\"ng-binding\"]')\n",

"for i in date:\n",

" Date.append(i.text)\n",

"print(Date)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "625cfa9a",

"metadata": {},

"outputs": [],

"source": [

"time=driver.find\_elements\_by\_xpath('//h5[@class=\"text-right ng-binding\"]')\n",

"for i in time:\n",

" if i.text is None :\n",

" Time.append(\"-\") \n",

" else:\n",

" Time.append(i.text)\n",

"print(Time)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "55878bc7",

"metadata": {},

"outputs": [],

"source": [

"#Checking the length of data extracted\n",

"print(len(match\_title),len(series),len(Date),len(Time),len(place))"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "5ab2aedc",

"metadata": {},

"outputs": [],

"source": [

"Youtube=pd.DataFrame({'MATCH TITLE':match\_title,'Series':series,'Date':Date,'Time':Time,'PLACE':place})\n",

"Youtube\n"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "b03be05f",

"metadata": {},

"outputs": [],

"source": [

"driver.close()"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "215604fb",

"metadata": {},

"outputs": [],

"source": [

"#Q.3\n",

"driver = webdriver.Chrome(\"chromedriver.exe\")\n",

"time.sleep(2)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "3074a68c",

"metadata": {},

"outputs": [],

"source": [

"driver.get('https://www.guru99.com/')"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "923a2b70",

"metadata": {},

"outputs": [],

"source": [

"sel\_button=driver.find\_element\_by\_xpath('//\*[@id=\"post-2669\"]/div/div[1]/div[1]/div[2]/div/div[2]/span[8]/a')\n",

"sel\_button.click()"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "4f73d90a",

"metadata": {},

"outputs": [],

"source": [

"tutorial\_button=driver.find\_element\_by\_xpath('//\*[@id=\"post-193\"]/div/div/table[5]/tbody/tr[34]/td[1]/a')\n",

"tutorial\_button.click()\n",

"time.sleep(2)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "cc4155b1",

"metadata": {},

"outputs": [],

"source": [

"Name=[]\n",

"Description=[]"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "bd2e0bd6",

"metadata": {},

"outputs": [],

"source": [

"name=driver.find\_elements\_by\_xpath('//td[1]')\n",

"for i in name:\n",

" Name.append(i.text)\n",

"print(Name)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "ee5e450d",

"metadata": {},

"outputs": [],

"source": [

"desc=driver.find\_elements\_by\_xpath('//td[2]')\n",

"for i in desc:\n",

" Description.append(i.text)\n",

"print(Description)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "a7f46e45",

"metadata": {},

"outputs": [],

"source": [

"print(len(Name),len(Description))"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "4f32b06e",

"metadata": {},

"outputs": [],

"source": [

"import pandas as pd\n",

"sel\_exception =pd.DataFrame({'Name':Name,'Description':Description})\n",

"sel\_exception"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "e5edd0c9",

"metadata": {},

"outputs": [],

"source": [

"driver.close()"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "2f211319",

"metadata": {},

"outputs": [],

"source": [

"#Q.4.........\n",

"\n",

"driver = webdriver.Chrome(\"chromedriver.exe\")\n",

"time.sleep(2)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "f8120187",

"metadata": {},

"outputs": [],

"source": [

"driver.get('http://statisticstimes.com/')\n",

"time.sleep(2)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "e78ab652",

"metadata": {},

"outputs": [],

"source": [

"dropbutton=driver.find\_element\_by\_xpath('//\*[@id=\"top\"]/div[2]/div[2]/button')\n",

"dropbutton.click()\n",

"time.sleep(2)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "16b064af",

"metadata": {},

"outputs": [],

"source": [

"economy=driver.find\_element\_by\_xpath(\"//div[@class='navbar']/div[2]/div/a[3]\") \n",

"try:\n",

" economy.click()\n",

"except ElementNotInteractableException:\n",

" driver.get(economy.get\_attribute('href'))\n",

"\n",

"time.sleep(2)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "72050caa",

"metadata": {},

"outputs": [],

"source": [

"gdp=driver.find\_element\_by\_xpath(\"/html/body/div[2]/div[2]/div[2]/ul/li[1]/a\")\n",

"try:\n",

" gdp.click()\n",

"except ElementNotInteractableException:\n",

" driver.get(gdp.get\_attribute('href')) "

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "806fbcbb",

"metadata": {},

"outputs": [],

"source": [

"Rank=[]\n",

"State=[]\n",

"GSDP17\_18=[]\n",

"GSDP18\_19=[]\n",

"GSDP19\_20=[]\n",

"Share2019=[]\n",

"GDP=[]"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "ff566372",

"metadata": {},

"outputs": [],

"source": [

"rank=driver.find\_elements\_by\_xpath(\"//table[@class='display dataTable']/tbody/tr/td[1]\")\n",

"for i in rank:\n",

" Rank.append(i.text)\n",

"\n",

"state=driver.find\_elements\_by\_xpath(\"//table[@class='display dataTable']/tbody/tr/td[2]\")\n",

"for i in state:\n",

" State.append(i.text)\n",

" \n",

"for i in driver.find\_elements\_by\_xpath(\"//table[@class='display dataTable']/tbody/tr/td[3]\"):\n",

" GSDP19\_20.append(i.text)\n",

" \n",

"for i in driver.find\_elements\_by\_xpath(\"//table[@class='display dataTable']/tbody/tr/td[4]\"):\n",

" GSDP18\_19.append(i.text)\n",

" \n",

"\n",

"for i in driver.find\_elements\_by\_xpath(\"//table[@class='display dataTable']/tbody/tr/td[5]\"):\n",

" Share2019.append(i.text)\n",

" \n",

"\n",

"for i in driver.find\_elements\_by\_xpath(\"//table[@class='display dataTable']/tbody/tr/td[6]\"):\n",

" GDP.append(i.text)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "fcd1248f",

"metadata": {},

"outputs": [],

"source": [

"#Creating a dataframe for the scrapped data\n",

"GDP\_data=pd.DataFrame({'Rank':Rank,'State':State,'GSDP(19-20)':GSDP19\_20,'GSDP(18-19)':GSDP18\_19,'Share2019':Share2019,\n",

" 'GDP($ billion)':GDP})\n",

"GDP\_data"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "6a30ce3c",

"metadata": {},

"outputs": [],

"source": [

"driver.close()"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "804e82c4",

"metadata": {},

"outputs": [],

"source": [

"#Q.5.......\n",

"driver = webdriver.Chrome(\"chromedriver.exe\")\n",

"time.sleep(2)\n",

"\n",

"driver.get('https://github.com/')\n",

"time.sleep(2)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "53c39f69",

"metadata": {},

"outputs": [],

"source": [

"button=driver.find\_element\_by\_xpath('/html/body/div[1]/header/div/div[2]/nav/ul/li[4]')\n",

"button.click()\n",

"time.sleep(2)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "c99df07d",

"metadata": {},

"outputs": [],

"source": [

"trending=driver.find\_element\_by\_xpath('/html/body/div[1]/header/div/div[2]/nav/ul/li[4]/details/div/ul/li[5]/a') \n",

"try:\n",

" trending.click()\n",

"except ElementNotInteractableException:\n",

" driver.get(trending.get\_attribute('href'))\n",

"time.sleep(2)\n"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "a03fe483",

"metadata": {},

"outputs": [],

"source": [

"rep\_title=[]\n",

"rep\_desc=[]\n",

"Count=[]\n",

"Lang=[]"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "87f86670",

"metadata": {},

"outputs": [],

"source": [

"title=driver.find\_elements\_by\_xpath(\"//h1[@class='h3 lh-condensed']/a\")\n",

"for i in title:\n",

" rep\_title.append(i.text)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "3d593042",

"metadata": {},

"outputs": [],

"source": [

" \n",

"\n",

"desc=driver.find\_elements\_by\_xpath('//p[@class=\"col-9 color-fg-muted my-1 pr-4\"]')\n",

"for i in desc:\n",

" if i.text is None:\n",

" rep\_desc.append('-')\n",

" else:\n",

" rep\_desc.append(i.text)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "3d43fdd8",

"metadata": {},

"outputs": [],

"source": [

" \n",

"count=driver.find\_elements\_by\_xpath('//div[@class=\"f6 color-fg-muted mt-2\"]//a[2]')\n",

"for i in count:\n",

" if i.text is None:\n",

" Count.append('-')\n",

" else:\n",

" Count.append(i.text)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "92216038",

"metadata": {},

"outputs": [],

"source": [

" \n",

"\n",

"lang=driver.find\_elements\_by\_xpath(\"//span[@class='d-inline-block ml-0 mr-3']\")\n",

"for i in lang:\n",

" if i.text is None:\n",

" Lang.append('-')\n",

" else:\n",

" Lang.append(i.text)\n",

" "

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "91d3ffb7",

"metadata": {},

"outputs": [],

"source": [

"print(len(rep\_title),len(rep\_desc),len(Count),len(Lang))"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "3b6085e8",

"metadata": {

"scrolled": true

},

"outputs": [],

"source": [

"github=pd.DataFrame({})\n",

"github['Repository title']=rep\_title[:18]\n",

"github['Description']=rep\_desc[:18]\n",

"github['Contributors count']=Count[:18]\n",

"github['Language used']=Lang\n",

"github"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "dae37586",

"metadata": {},

"outputs": [],

"source": [

"driver.close()"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "de2749e5",

"metadata": {},

"outputs": [],

"source": [

"#Q.6.......\n",

"driver = webdriver.Chrome(\"chromedriver.exe\")\n",

"time.sleep(2)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "2705199b",

"metadata": {},

"outputs": [],

"source": [

"driver.get('https://www.billboard.com/')"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "12bd7b6b",

"metadata": {},

"outputs": [],

"source": [

"Song\_name=[]\n",

"Artist\_name=[]\n",

"Last\_week\_rank=[]\n",

"Peak\_rank=[]\n",

"Weeks\_on\_board=[]"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "c1a903b2",

"metadata": {},

"outputs": [],

"source": [

"hot\_100=driver.find\_element\_by\_xpath(\"//nav[@class='header\_\_subnav bg--light']/ul/li[3]/a\")\n",

"try:\n",

" hot\_100.click()\n",

"except ElementNotInteractableException:\n",

" driver.get(hot\_100.get\_attribute('href'))"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "3022e2d3",

"metadata": {},

"outputs": [],

"source": [

"Song=[]\n",

"Artist=[]\n",

"Lastweek=[]\n",

"Peak=[]\n",

"Weeks\_on\_board=[]"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "6b00685a",

"metadata": {},

"outputs": [],

"source": [

"song=driver.find\_elements\_by\_xpath(\"//span[@class='chart-element\_\_information\_\_song text--truncate color--primary']\")\n",

"for i in song:\n",

" Song.append(i.text)\n",

" \n",

"\n",

"artist=driver.find\_elements\_by\_xpath(\"//span[@class='chart-element\_\_information\_\_artist text--truncate color--secondary']\")\n",

"for i in artist:\n",

" Artist.append(i.text)\n",

" \n",

"\n",

"last\_week=driver.find\_elements\_by\_xpath(\"//div[@class='chart-element\_\_meta text--center color--secondary text--last']\")\n",

"for i in last\_week:\n",

" if i.text is None:\n",

" Lastweek.append('-')\n",

" else:\n",

" Lastweek.append(i.text)\n",

" \n",

"\n",

"peak=driver.find\_elements\_by\_xpath(\"//div[@class='chart-element\_\_meta text--center color--secondary text--peak']\")\n",

"for i in peak:\n",

" if i.text is None:\n",

" Peak.append('-')\n",

" else:\n",

" Peak.append(i.text)\n",

" \n",

"\n",

"weeks=driver.find\_elements\_by\_xpath(\"//div[@class='chart-element\_\_meta text--center color--secondary text--week']\")\n",

"for i in weeks:\n",

" if i.text is None:\n",

" Weeks\_on\_board.append('-')\n",

" else:\n",

" Weeks\_on\_board.append(i.text)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "e797b9ff",

"metadata": {},

"outputs": [],

"source": [

"billboard=pd.DataFrame({'Song':Song,'Artist':Artist,'Last week Rank':Lastweek,'Peak Rank':Peak,'Weeks on board':Weeks\_on\_board})\n",

"billboard"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "76ec0e98",

"metadata": {},

"outputs": [],

"source": [

"# Q.7.............\n",

" \n",

"driver=webdriver.Chrome(\"chromedriver.exe\")\n",

"driver.get('https://www.naukri.com/')"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "df548f66",

"metadata": {},

"outputs": [],

"source": [

"companies=driver.find\_element\_by\_xpath('//\*[@id=\"root\"]/div[2]/div[1]/div/ul/li[2]/a/div')\n",

"companies.click()"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "0e47fe27",

"metadata": {},

"outputs": [],

"source": [

"recruit=driver.find\_element\_by\_xpath('//div[@class=\"mTxt\"]')\n",

"recruit.click()"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "d42c484c",

"metadata": {},

"outputs": [],

"source": [

"search\_job=driver.find\_element\_by\_xpath(\"//input[@class='sugInp']\")\n",

"search\_job.send\_keys('Data science') "

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "eb5cb72f",

"metadata": {},

"outputs": [],

"source": [

"driver.find\_element\_by\_xpath('//\*[@id=\"qsbFormBtn\"]').click()"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "b0ffc754",

"metadata": {},

"outputs": [],

"source": [

"Name=[]\n",

"Designation=[]\n",

"Company=[]\n",

"Skills=[]\n",

"Location=[]"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "4a2a2c6a",

"metadata": {},

"outputs": [],

"source": [

"name=driver.find\_elements\_by\_xpath('//a[@class=\"title fw500 ellipsis\"]')\n",

"for i in name:\n",

" Name.append(i.text)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "2a91308f",

"metadata": {},

"outputs": [],

"source": [

" \n",

"desc=driver.find\_elements\_by\_xpath(\"//span[@class='ellipsis clr']\")\n",

"for i in desc:\n",

" Designation.append(i.text)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "2336733d",

"metadata": {},

"outputs": [],

"source": [

"company=driver.find\_elements\_by\_xpath(\"//p[@class='highlightable']/a[2]\")\n",

"for i in company:\n",

" Company.append(i.text)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "e663b903",

"metadata": {},

"outputs": [],

"source": [

"skills=driver.find\_elements\_by\_xpath(\"//div[@class='hireSec highlightable']\")\n",

"for i in skills:\n",

" Skills.append(i.text)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "07a7cce3",

"metadata": {},

"outputs": [],

"source": [

"location=driver.find\_elements\_by\_xpath(\"//small[@class='ellipsis']\")\n",

"for i in location:\n",

" Location.append(i.text) "

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "d3c11ef5",

"metadata": {},

"outputs": [],

"source": [

"print(len(Name),len(Designation),len(Company),len(Skills),len(Location))\n"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "c86e4e0b",

"metadata": {},

"outputs": [],

"source": [

"naukri=pd.DataFrame({})\n",

"naukri['Name']=Name[:48]\n",

"naukri['Designation']=Designation[:48]\n",

"naukri['Company']=Company[:48]\n",

"naukri['Skills']=Skills[:48]\n",

"naukri['Location']=Location\n",

"naukri"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "c0bebdfb",

"metadata": {},

"outputs": [],

"source": [

"#Q.8.................\n",

"driver=webdriver.Chrome(\"chromedriver.exe\") \n",

"driver.get('https://www.theguardian.com/news/datablog/2012/aug/09/best-selling-books-all-time-fifty-shades-grey-compare/')"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "02bbae30",

"metadata": {},

"outputs": [],

"source": [

"Book=[]\n",

"Author=[]\n",

"Volume=[]\n",

"Publisher=[]\n",

"Genre=[]"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "570ac015",

"metadata": {},

"outputs": [],

"source": [

"book=driver.find\_elements\_by\_xpath(\"//table[@class='in-article sortable']/tbody/tr/td[2]\")\n",

"for i in book:\n",

" Book.append(i.text)\n",

" \n",

"\n",

"author=driver.find\_elements\_by\_xpath(\"//table[@class='in-article sortable']/tbody/tr/td[3]\")\n",

"for i in author:\n",

" Author.append(i.text)\n",

" \n",

"\n",

"volume=driver.find\_elements\_by\_xpath(\"//table[@class='in-article sortable']/tbody/tr/td[4]\")\n",

"for i in volume:\n",

" Volume.append(i.text)\n",

" \n",

"publisher=driver.find\_elements\_by\_xpath(\"//table[@class='in-article sortable']/tbody/tr/td[5]\")\n",

"for i in publisher:\n",

" Publisher.append(i.text)\n",

" \n",

"\n",

"genre=driver.find\_elements\_by\_xpath(\"//table[@class='in-article sortable']/tbody/tr/td[6]\")\n",

"for i in genre:\n",

" Genre.append(i.text)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "d163ed62",

"metadata": {},

"outputs": [],

"source": [

"novels=pd.DataFrame({})\n",

"novels['Book']=Book\n",

"novels['Author']=Author\n",

"novels['Volumes sold']=Volume\n",

"novels['Publisher']=Publisher\n",

"novels['Genre']=Genre\n",

"novels"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "74214864",

"metadata": {},

"outputs": [],

"source": [

"#Q.9............\n",

"driver=webdriver.Chrome(\"chromedriver.exe\") \n",

"driver.get('https://www.imdb.com/list/ls095964455/')"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "f82e0cec",

"metadata": {},

"outputs": [],

"source": [

"Name=[]\n",

"Year=[]\n",

"Genre=[]\n",

"Runtime=[]\n",

"Rating=[]\n",

"Vote=[]"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "0cc40d71",

"metadata": {},

"outputs": [],

"source": [

"name=driver.find\_elements\_by\_xpath(\"//h3[@class='lister-item-header']/a\")\n",

"for i in name:\n",

" Name.append(i.text)\n",

" \n",

"\n",

"year=driver.find\_elements\_by\_xpath(\"//span[@class='lister-item-year text-muted unbold']\")\n",

"for i in year:\n",

" Year.append(i.text)\n",

" \n",

"\n",

"genre=driver.find\_elements\_by\_xpath(\"//span[@class='genre']\")\n",

"for i in genre:\n",

" Genre.append(i.text)\n",

" \n",

"\n",

"runtime=driver.find\_elements\_by\_xpath(\"//span[@class='runtime']\")\n",

"for i in runtime:\n",

" Runtime.append(i.text)\n",

" \n",

"\n",

"rating=driver.find\_elements\_by\_xpath(\"//div[@class='ipl-rating\n",

"for i in rating:\n",

" Rating.append(i.text)\n",

" \n",

"votes=driver.find\_elements\_by\_xpath(\"//span[@name='nv']\")\n",

"for i in votes:\n",

" Vote.append(i.text)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "dcebe913",

"metadata": {},

"outputs": [],

"source": [

"#Creating a dataframe for the scrapped data\n",

"IMDB=pd.DataFrame({'Name':Name,'Year span':Year,'Genre':Genre,'Runtime':Runtime,'Rating':Rating,'Votes':Vote})\n",

"IMDB \n",

" "

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "2cbb5daf",

"metadata": {},

"outputs": [],

"source": [

"#Q.10..........\n",

"driver=webdriver.Chrome(\"chromedriver.exe\") \n",

"driver.get('https://archive.ics.uci.edu/')"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "dd5c7cbb",

"metadata": {},

"outputs": [],

"source": [

"datasets=driver.find\_element\_by\_xpath(\"/html/body/table[1]/tbody/tr/td[2]/span[2]/a/font/b\")\n",

"try:\n",

" datasets.click()\n",

"except ElementNotInteractableException:\n",

" driver.get(datasets.get\_attribute('href'))"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "7c348082",

"metadata": {},

"outputs": [],

"source": [

"Type=[]\n",

"Task=[]\n",

"Attribute=[]\n",

"No\_of\_Instance=[]\n",

"No\_of\_Attribute=[]\n",

"Year=[]"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "9d822aea",

"metadata": {},

"outputs": [],

"source": [

"try:\n",

" names=driver.find\_elements\_by\_xpath(\"/html/body/table[2]/tbody/tr/td[2]/table[2]/tbody/tr/td[1]/table/tbody/tr/td[2]/p/b/a\")\n",

" for i in names:\n",

" Name.append(i.text)\n",

"except NoSuchElementException:\n",

" Name.append('-')\n",

" \n",

"try:\n",

"types=driver.find\_elements\_by\_xpath(\"/html/body/table[2]/tbody/tr/td[2]/table[2]/tbody/tr/td[2]\")\n",

" for i in types[1:]:\n",

" Type.append(i.text)\n",

"except NoSuchElementException:\n",

" Type.append('-')\n",

" \n",

"try:\n",

" task=driver.find\_elements\_by\_xpath(\"/html/body/table[2]/tbody/tr/td[2]/table[2]/tbody/tr/td[3]\")\n",

" for i in task[1:]:\n",

" Task.append(i.text)\n",

"except NoSuchElementException:\n",

" Task.append('-')\n",

" \n",

"try:\n",

" attribute=driver.find\_elements\_by\_xpath(\"/html/body/table[2]/tbody/tr/td[2]/table[2]/tbody/tr/td[4]\")\n",

" for i in attribute[1:]:\n",

" Attribute.append(i.text)\n",

"except NoSuchElementException:\n",

" Attribute.append('-')\n",

"\n",

"try:\n",

" instance=driver.find\_elements\_by\_xpath(\"/html/body/table[2]/tbody/tr/td[2]/table[2]/tbody/tr/td[5]\")\n",

" for i in instance[1:]:\n",

" No\_of\_Instance.append(i.text)\n",

"except NoSuchElementException:\n",

" No\_of\_Instance.append('-')\n",

" \n",

"\n",

"try:\n",

" attribute\_no=driver.find\_elements\_by\_xpath(\"/html/body/table[2]/tbody/tr/td[2]/table[2]/tbody/tr/td[6]\")\n",

" for i in attribute\_no[1:]:\n",

" No\_of\_Attribute.append(i.text)\n",

"except NoSuchElementException:\n",

" No\_of\_Attribute.append('-')\n",

" \n",

"try:\n",

" year=driver.find\_elements\_by\_xpath(\"/html/body/table[2]/tbody/tr/td[2]/table[2]/tbody/tr/td[7]\")\n",

" for i in year[1:]:\n",

" Year.append(i.text)\n",

"except NoSuchElementException:\n",

" Year.append('-')"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "8981cd58",

"metadata": {},

"outputs": [],

"source": [

"#Creating the dataframe\n",

"df=pd.DataFrame({\"Name\":Name,\"Data types\":Type,\"Default Task\":Task,\"Attribute types\":Attribute, \n",

" \"No of instances\":No\_of\_Instance,\"No of atrributes\":No\_of\_Attribute,\"Year\":Year})\n",

"df"

]

}

],

"metadata": {

"kernelspec": {

"display\_name": "Python 3 (ipykernel)",

"language": "python",

"name": "python3"

},

"language\_info": {

"codemirror\_mode": {

"name": "ipython",

"version": 3

},

"file\_extension": ".py",

"mimetype": "text/x-python",

"name": "python",

"nbconvert\_exporter": "python",

"pygments\_lexer": "ipython3",

"version": "3.9.7"

}

},

"nbformat": 4,

"nbformat\_minor": 5

}