**AppingWay Technologies**

**Name: BRIJ ROKAD E-mail Id: brij.rokad@gmail.com**

**Python Task 1:** Given two lists L1 = [‘a’, ’b’, ’c’], L2 = [‘b’, ‘d’], find common elements, find elements present in L1 and not in L2?

**Code:**

**L1 = ['a','b','c']**

**L2 = ['b','d']**

**def intersection(L1,L2):**

**return list(set(L1) & set(L2))**

## intersection() is a operator which returns the common element of two sets

**print ("Common Elements in list L1 and L2:",intersection(L1,L2))**

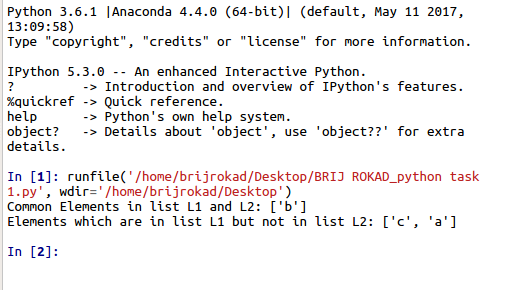
**x = list(set(L1) - set(L2))**

**print ("Elements which are in list L1 but not in list L2:",x)**

## here two sets will be compared and if elements are found common, then they

## will be removed from the set L1 and remaining elements wil be returned

**Output:**



**Python Task 2:** How many Thursdays were there between 1990 - 2000?

**Code:**

**import calendar**

**import datetime**

**def weekday\_count(s,e):**

**s\_date = datetime.datetime.strptime(s,'%d/%m/%Y')**##assigning strat and end date

**e\_date = datetime.datetime.strptime(e,'%d/%m/%Y')**##using datetime using datetime library

**week = {}**

**for i in range((e\_date-s\_date).days):**

**day = calendar.day\_name[(s\_date + datetime.timedelta(days=i+1)).weekday()]**

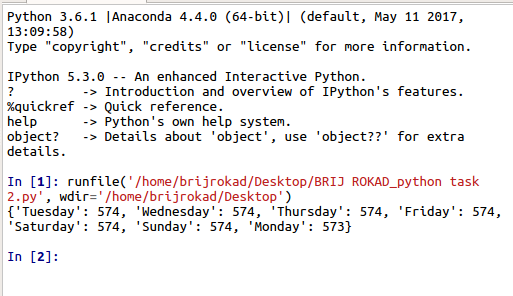
**week[day] = week[day] + 1 if day in week else 1**

**return week** ##counting number of weeks, every week will add one number of day

## if week is not complete then number of days will be returned

**print(weekday\_count("01/01/1990", "31/12/2000"))**

**Output:**

****

**Javascript task:** Given the following Javascript array:

var data = [0, 1, 2, ‘stop’, 2, 0, 1, ‘stop’]

write a Javascript function or expression that returns an array with just the zeroes removed.

**Code:**

**var data = [0,1,2,"stop",2,0,1,"stop"];**

**function remove(array,element){**

**for(var i = array.length-1; i--;){**

**if(array[i] === element)** // It will check the entire array

**array.splice(i, 1);** // If zero found it will remove it

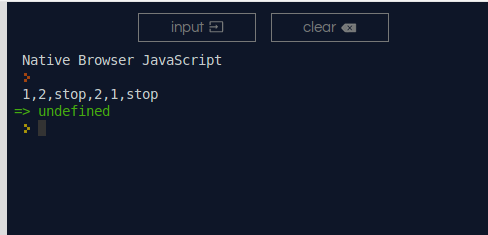
**}** // splice is pre-define function

**}**

**remove(data,0)** // function call

**console.log(data.toString());**

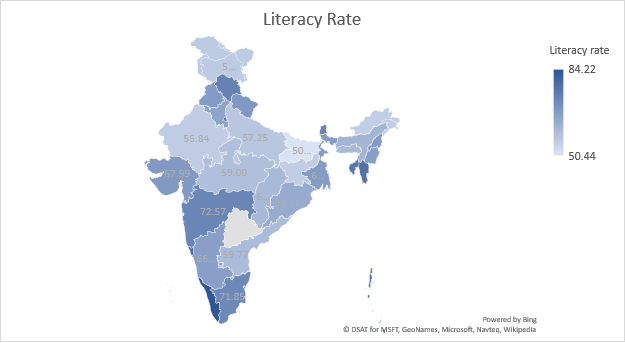
**Output:**

****

**Analysis task: Use case 2 - 2011 India Census**

**Question 1:** Create a geographic map of states with low literacy rates.

**Ans:** I have uploaded Excel file name "Use case 2 - 2011 India Census\_Q1.xlsx", which has table of all the state literacy rate Geographical Map of it. Literacy rate ranging from 50.44% to 84.22%



**Question 3:** How does the mobile penetration vary in regions (districts or states) with high or low agricultural workers?

**Ans:** By analyzing the given data one can interpret that the state with less mobile penetration are low on agriculture workers. On the other hand, with more mobile penetration states have high number of agriculture workers. Excel file name “Use case 2 - 2011 India Census\_Q1.xlsx”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| District code | State name | Agricultural\_Workers | H\_TM\_P\_L\_only | H\_TM\_P\_M\_only |
| 55 | CHANDIGARH | 1687 | 14027 | 148411 |
| 23 | HIMACHAL PRADESH | 175038 | 109706 | 907586 |
| 56 | UTTARAKHAND | 403301 | 63362 | 1294045 |
| 1 | JAMMU & KASHMIR | 547705 | 72820 | 1194854 |
| 69 | HARYANA | 1528133 | 210845 | 3157367 |
| 35 | PUNJAB | 1588455 | 361086 | 3368428 |

(Sample Example of Worker VS Mobile)