**File I/O**

1. Open the file sample1.txt and read the first 25 characters
2. Append the sentence “I am learning Python” to the file
3. Create a new file sample2.txt and write “Cricket is India’s favorite game”

**Functions**

1. Write a function that calculates the weekly pay by taking in two inputs – Hours worked in the week & pay per hour from the user. Test out your function using 10/hr as pay & 40 as number of hours worked
2. Write a function that calculates your BMI given input of height (in cm) & weight (in Kg).

Calculation – BMI = (wt / ht\*ht)\*10000

**Conditional Stmts**

1. Take the year as an input from a user and check if it is a leap year or not. Leap year is one which is divisible by 4 but not by 100 unless it is divisible by 400
2. Rewrite the pay computation such that for all hours worked more than 40, the pay is 1.5 times the per hour pay. Also put in checks to ensure that pay per hour is not more than 15 and hours worked is not more than 75. Put in appropriate error stmts
3. Write a program to get temperature input (number & unit of temp i.e. farenheit or centigrade) from the user and convert from Farenheit to Centigrade or vice versa

Tc = (5/9)\*(Tf-32)

Tf = (9/5)\*Tc + 32

1. Write a function to calculate the factorial of a given number

**Loops**

1. Print x number of terms of the Fibonacci series. Take input from user for number of terms

1, 1, 2, 3, 5, 8, 13, 21....

1. Print all prime numbers from 1 to 50
2. Find the list of Pythagorean numbers. Take max value to be used from user.
3. Count the number of lines in the file “example\_text.txt

**Strings**

1. Given two strings of different lengths, print an output in the form short-long-short

**e.g. Hi, Hello  hihellohi**

1. Given a string “TEMP”, write a code that will give an output TTETEMTEMP
2. Given two strings check if either of the strings appears at the very end of the other string. If yes, print “Yes, *string XXX* is at the end of *string YYY*". If not, print “No, the two strings are independent”. This should work regardless of case i.e. not case sensitive
3. Write a code to identify a palindrome
4. From the file “txt\_mbox-short.txt’ extract all the host names from the different emails

**Lists**

1. Given a list of integers [1,5,100,56,10,8,4,9,12,21,34,55,89], create a new list which contains all members of the list whose value is less than 15
2. Given a list of integers, calculate the average of all the elements in the list ignoring outliers. In this case, assume outliers are the largest and smallest numbers in the list (if there are 2 instances of the largest or smallest number, ignore one of them)
3. Take two lists:

a = [1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89]

b = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13]

write a program that returns a list that contains only the elements that are common between the lists (without duplicates).

1. From the text file “example\_text.txt, count the number of unique **words**. Print the list of unique words

**Dictionaries**

1. **Create two dictionaries for a store one containing the stock (quantity) of the items and one with the price for each item. Use the below info**

**prices = {'Bread':20, 'Milk': 10, 'Eggs': 15, "Vegetables": 30}**

**stock = {'Bread':10, 'Milk': 20, 'Eggs': 60, 'Vegetables': 25}**

* 1. **Get input from the user on the item (assume they ask for only 1 item) they want to buy and the quantity. Print out the total bill for the purchase. If stock is less than amount they want to buy, print ”Out of Stock”**
  2. **Repeat step (a). In addition to total bill for the purchase, also update the stock to reflect the quantity that has been bought and print out the remaining stock**
  3. **Modify the code above to get a list of items from the user as against a single item**

1. **From the text file “example\_text.txt, identify all the unique words and count the number of times they have occurred.**
2. **Write a program to read through the text file “mbox-short.txt” and figure out from which email the maximum messages have come. Hint: Look for lines starting with the ‘From ‘ text and take the second word as the name of the person who has sent the message**
3. **Write a program to read through the text file “mbox-short.txt” and figure out the distribution by hour of the day for all the messages sent. Hint: Look for lines starting with the “From “ text and see where the time comes**

**WebScraping**

**Go to the link 'https://en.wikipedia.org/wiki/2016\_Summer\_Olympics' and extract the first 25 links from this page. Then go to the last link from this set, get the first 25 links again and print the last 5 links**

**Pandas**

**Using the loan\_data.csvs dataset, do the following**

1. **Read the file and find the number of rows & columns in the data set as well as understand the datatype of the various columns**
2. **Determine the frequency distribution of the field property area**
3. **Calculate the summary stats of the numeric variables in the data set**
4. **Create a new dataframe which only has data for Graduates and who have been approved for a loan**
5. **Determine if the credit history of an applicant has a relationship to loan application approval (use a pandas pivot table or cross tab)**
6. **Determine the number of missing values in the dataset by variable**
7. **Impute the missing values for loan amount using the median value of loan amount**
8. **Find the mean of sum of applicant and co applicant income. Change loan status to 'N' for all applicants who fall below the mean.**