Extract Valid Emails

```
import re

def validate_emails(emails):
    """
    Validate a list of email addresses and return only the valid ones.

    Parameters:
    emails (list): A list of email addresses to validate.

    Returns:
    list: A list containing only the valid email addresses.
    """
    ans = [] #gaur.11dec @ gmail. com
    pattern=r'^[\w\.\-]+@[a-zA-Z0-9\-]+\.[a-zA-Z]{2,}$'
    for email in emails:
        if re.match(pattern,email):
            ans.append(email)

    return ans
```

String Manipulation

```
def modify(s):
    '''Input: s is the string
      Output:return the resultant string with described modifications'''
    # YOUR CODE GOES HERE
    if len(s)<3:
        return s


    if s[-3:]=='ing': #[start:end:step] -> start=-3, end=last char, step=+1
        return s+'ly'


    return s+'ing'
```

Adding common keys

```
def commonKey(dict1, dict2):
 dict3 = {}
```

```
#YOUR CODE GOES HERE
for key in dict1:
    if key in dict2:
        dict3[key] = dict1[key] + dict2[key]
print(dict3)
```

Which regex matches any lowercase letter except 'a'?
^a

Different uses of ^
[^a] -> not
^[a] -> starts with

re.match("hello", string) -> searches only the start of the string

re.search("hello", string) -> searches anywhere in the string

Break: 9:50 - 10:00 pm

Flames

```
def flames_game(boy, girl):
  #clean the data
  boy=boy.replace(" ","").lower() #aakar
  girl=girl.replace(" ","").lower() #disha

  #get the unique characters -> by using set
  boy_set=set(boy) #{a,k,r}
  girl_set=set(girl) #{d,i,s,h,a}

  #remove common
  common = boy_set & girl_set #gives common characters -> a
  boy_unique=boy_set - common #{k,r}
  girl_unique=girl_set - common #{d,i,s,h}

  #total count
  score=len(boy_unique)+len(girl_unique) # 2+4=> 6

  flames=["F","L","A","M","E","S"]

  index=(score-1)%len(flames)
  result_letter=flames[index]

  relation_dict={'F':"Friends","L":"Love",
                "A":"Affection","M":"Marriage",
                "E":"Enemy","S":"Sibling"}

  return relation_dict[result_letter]

boy=input("Boy's name")
girl=input("Girl's name")
print("the relationship is",flames_game(boy,girl))
```

Length of unique words

```
def set_operation(sent1,sent2):
    ''' input:sent1,sent2-two sentences taken as inputs
         output:return the sum of length of unique words.'''

    # YOUR CODE GOES HERE
    sen1_words=sent1.split()
    sen2_words=sent2.split()
    a=set(sen1_words)
    b=set(sen2_words)
    return (len(a)+len(b))
```

Inverted triangle

```
def main():
    # YOUR CODE GOES HERE
    # Please take input and print output to standard input/output (stdin/stdout)
    # E.g. 'input()/raw_input()' for input & 'print' for output
    n=int(input()) #4
    for r in range(n): # 0, 1, 2, 3
        for c in range(n-r): # 0-> 0
            print(c+1, end="")  # 1
            if (c!=n-r-1):
                print(" ",end="")
        print()

    return 0

if __name__ == '__main__':
    main()
    # 1 2 3 4
    # 1 2 3
    # 1 2
    # 1
```