Task1:

You need to write a function in python that return all such numbers which are divisible by 7 but are not a multiple of 5, between 2000 and 3200 (both included). Return the obtained numbers in a comma-separated sequence on a single line.

```
In [38]: def divisible_7 ():
result
[]
for i in range (2000, 3201):
if i % 7 == 0 and i % 5 != 0:
result.append(i)
return result
```

```
In [79]: print(divisible_7())
```

```
[2002, 2009, 2016, 2023, 2037, 2044, 2051, 2058, 2072, 2079, 2086, 2093, 2107, 2114,
2121, 2128, 2142, 214 9, 2156, 2163, 2177, 2184, 2191, 2198, 2212, 2219, 2226, 2233, 2247,
2254, 2261, 2268, 2282, 2289, 2296, 23 03, 2317, 2324, 2331, 2338, 2352, 2359, 2366, 2373, 2387,
2394, 2401, 2408, 2422, 2429, 2436, 2443, 2457, 2 464, 2471, 2478, 2492, 2499, 2506, 2513, 2527, 2534,
2541, 2548, 2562, 2569, 2576, 2583, 2597, 2604, 2611, 2618, 2632, 2639, 2646, 2653, 2667, 2674, 2681,
2688, 2702, 2709, 2716, 2723, 2737, 2744, 2751, 2758, 2772, 2779, 2786, 2793, 2807, 2814, 2821, 2828,
2842, 2849, 2856, 2863, 2877, 2884, 2891, 2898, 2912, 2919, 2926, 2933, 2947, 2954, 2961, 2968, 2982,
2989, 2996, 3003, 3017, 3024, 3031, 3038, 3052, 3059, 3066, 3073, 3087, 3094, 3101, 3108, 3122, 3129,
3136, 3143, 3157, 3164, 3171, 3178, 3192, 3199]
```

Task2:

You need code a function that calculates and returns the value according to the given formula:P = Square root of [(2 * A * B)/C] Following are the fixed values of A and B: A is 50. B is 30. The values of the literal C should be taken as console input to your program in a comma-separated sequence.

Example

Let us assume the following comma separated input sequence is given to the program: 100,150,180 The output of the program should be: 18,22,24

```
In [51]: def cal_values ():
A = 50
B = 30
C_values = input("Enter values of C seperated by comma :")
C_1st =
C_values.split(",")
result = []
for C in C_1st:
C = int(C)
P = ((2 * A * B)/C) ** 1/2
result.append (int(P))
return result
cal_values()

Enter values of C seperated by comma : 5,3,2
,2

Out [51]: [300, 500, 750]
```

Task3:

You need to write a function that takes a comma separated sequence of words as input and prints the words in a comma-separated sequence after sorting them alphabetically. Suppose the following input is supplied to the program: without, hello,bag,world Then, the output should be: bag,hello, without, world

```
In [56]: def words ():
words= input("Enter words seperated by comma: ") word_lst
=
words.split(",")
word_1st.sort()
print(", ".join(word_lst))
words()

Enter words seperated by comma: hello, without, bag, world

bag, hello, without, world
```

Task4:

You need to write a program that takes sequence of lines as input and prints the lines after making all characters in the sentence capitalized. Suppose the following input is supplied to the program: Hello world,Practice makes perfect

Then, the output should be: HELLO WORLD,PRACTICE MAKES PERFECT

```
In [1]: def capital_sent():
lines = []
while True:
line input("Enter a line or press enter to finish :")
"
if not line:
break
lines.append (line.upper() )
for line in lines:
print(line)
capital_sent()
```

```
Enter a line or press enter to finish: hello world
Enter a line or press enter to finish practice makes perfect
Enter a line or press enter to finish :

HELLO WORLD
PRACTICE MAKES PERFECT
```

Task5:

You need to write a function that counts the number of vowels in a given sentence as input from console. Suppose the following input is supplied to the program: Hello world,Practice makes perfect

Then, the output should be: a appeared 2 times,e appeared 5 times,i appeared 1 time,o appeared 2 times,u appeared 0 time PRACTICE MAKES PERFECT

```
In [70]: def count_vowels():
sent = input("Enter a sentence: "). lower()
```

```
vowels = "aeiou"
for vowel in vowels:
print (f" {vowel} appeared {sent.count(vowel)} times")
count_vowels()
```

```
Enter a sentence: Hello World, Practice Makes Perfect

a appeared 2 times
e appeared 5 times
i appeared 1 times
o appeared 2 times u appeared 0 times
```

Task6:

You need write a function that traces and makes a list of all such numbers from 1000 to 3000 in which all the digits are even numbers.

```
In [81]: def num_lst():
lst=[]
for i in range (1000, 3001):
if i % 2 == 0:
lst.append(i)
return lst
print(num_lst())
```

```
[1000, 1002, 1004, 1006, 1008, 1010, 1012, 1014, 1016, 1018, 1020, 1022, 1024, 1026, 1028,
1030, 1032, 103 4, 1036, 1038, 1040, 1042, 1044, 1046, 1048, 1050, 1052, 1054, 1056, 1058, 1060, 1062,
1064, 1066, 1068, 10 70, 1072, 1074, 1076, 1078, 1080, 1082, 1084, 1086, 1088, 1090, 1092, 1094,
1096, 1098, 1100, 1102, 1104, 1 106, 1108, 1110, 1112, 1114, 1116, 1118, 1120, 1122, 1124, 1126,
1128, 1130, 1132, 1134, 1136, 1138, 1140, 1142, 1144, 1146, 1148, 1150, 1152, 1154, 1156, 1158, 1160,
1162, 1164, 1166, 1168, 1170, 1172, 1174, 1176, 1178, 1180, 1182, 1184, 1186, 1188, 1190, 1192,
1194, 1196, 1198, 1200, 1202, 1204, 1206, 1208, 1210, 1212, 1214, 1216, 1218, 1220, 1222, 1224, 1226,
1228, 1230, 1232, 1234, 1236, 1238, 1240, 1242, 1244, 1246, 1248, 1250, 1252, 1254, 1256, 1258, 1260,
1262, 1264, 1266, 1268, 1270, 1272, 1274, 1276, 1278, 1280, 1282, 1284, 1286, 1288, 1290, 1292,
1294, 1296, 1298, 1300, 1302, 1304, 1306, 1308, 1310, 1312, 1314, 1316, 1318, 1320, 1322,
1324, 1326, 1328, 1330, 1332, 1334, 1336, 1338, 1340, 1342, 1344, 1346, 1348, 1350, 1352, 1354, 1356, 1358,
1360, 1362, 1364, 1366, 1368, 1370, 1372, 1374, 1376, 1378, 1380, 1382, 1384, 1386, 1388, 1390, 1392, 1394,
1396, 1398, 1400, 1402, 1404, 1406, 1408, 1410, 1412, 1414, 1416, 1418, 1420, 1422, 1424, 1426, 1428,
1430, 1432, 1434, 1436, 1438, 1440, 1442, 1444, 1446, 1448, 1450, 1452, 1454, 1456, 1458,
1460, 1462, 1464, 1466, 1468, 1470, 1472, 1474, 1476, 1478, 1480, 1482, 1484, 1486, 1488, 1490,
1492, 1494, 1496, 1498, 1500, 1502, 1504, 1506, 1508, 1510, 1512, 1514, 1516, 1518, 1520, 1522,
1524, 1526, 1528, 1530, 1532, 1534, 1536, 1538, 1540, 1542, 1544, 1546, 1548, 1550, 1552, 1554, 1556, 1558,
1560, 1562, 1564, 1566, 1568, 1570, 1572, 1574, 1576, 1578, 1580, 1582, 1584, 1586, 1588, 1590, 1592, 1594,
1596, 1598, 1600, 1602, 1604, 1606, 1608, 1610, 1612, 1614, 1616, 1618, 1620, 1622, 1624, 1626, 1628, 1630,
1632, 1634, 1636, 1638, 1640, 1642, 1644, 1646, 1648, 1650, 1652, 1654, 1656, 1658, 1660, 1662, 1664,
1666, 1668, 1670, 1672, 1674, 1676, 1678, 1680, 1682, 1684, 1686, 1688, 1690, 1692, 1694, 1696, 1698,
1700, 1702, 1704, 1706, 1708, 1710, 1712, 1714, 1716, 1718, 1720, 1722, 1724, 1726, 1728, 1730, 1732,
1734, 1736, 1738, 1740, 1742, 1744, 1746, 1748, 1750, 1752, 1754, 1756, 1758, 1760, 1762, 1764, 1766,
```

1768, 1770, 1772, 1774, 1776, 1778, 1780, 1782, 1784, 1786, 1788, 1790, 1792, 1794, 1796, 1798, 1800, 1802, 1804, 1806, 1808, 1810, 1812, 1814, 1816, 1818, 1820, 1822, 1824, 1826, 1828, 1830, 1832, 1834, 1836, 1838, 1840, 1842, 1844, 1846, 1848, 1850, 1852, 1854, 1856, 1858, 1860, 1862, 1864, 1866, 1868, 1870, 1872, 1874, 1876, 1878, 1880, 1882, 1884, 1886, 1888, 1890, 1892, 1894, 1896, 1898, 1900, 1902, 1904, 1906, 1908, 1910, 1912, 1914, 1916, 1918, 1920, 1922, 1924, 1926, 1928, 1930, 1932, 1934, 1936, 1938, 1940, 1942, 1944, 1946, 1948, 1950, 1952, 1954, 1956, 1958, 1960, 1962, 1964, 1966, 1968, 1970, 1972, 1974, 1976, 1978, 1980, 1982, 1984, 1986, 1988, 1990, 1992, 1994, 1996, 1998, 2000, 2002, 2004, 2006, 2008, 2010, 2012, 2014, 2016, 2018, 2020, 2022, 2024, 2026, 2028, 2030, 2032, 2034, 2036, 2038, 2040, 2042, 2044, 2046, 2048, 2050, 2052, 2054, 2056, 2058, 2060, 2062, 2064, 2066, 2068, 2070, 2072, 2074, 2076, 2078, 2080, 2082, 2084, 2086, 2088, 2090, 2092, 2094, 2096, 2098, 2100, 2102, 2104, 2106, 2108, 2110, 2112, 2114, 2116, 2118, 2120, 2122, 2124, 2126, 2128, 2130, 2132, 2134, 2136, 2138, 2140, 2142, 2144, 2146, 2148, 2150, 2152, 2154, 2156, 2158, 2160, 2162, 2164, 2166, 2168, 2170, 2172, 2174, 2176, 2178, 2180, 2182, 2184, 2186, 2188, 2190, 2192, 2194, 2196, 2198, 2200, 2202, 2204, 2206, 2208, 2210, 2212, 2214, 2216, 2218, 2220, 2222, 2224, 2226, 2228, 2230, 2232, 2234, 2236, 2238, 2240, 2242, 2244, 2246, 2248, 2250, 2252, 2254, 2256, 2258, 2260, 2262, 2264, 2266, 2268, 2270, 2272, 2274, 2276, 2278, 2280, 2282, 2284, 2286, 2288, 2290, 2292, 2294, 2296, 2298, 2300, 2302, 2304, 2306, 2308, 2310, 2312, 2314, 2316, 2318, 2320, 2322, 2324, 2326, 2328, 2330, 2332, 2334, 2336, 2338, 2340, 2342, 2344, 2346, 2348, 2350, 2352, 2354, 2356, 2358, 2360, 2362, 2364, 2366, 2368, 2370, 2372, 2374, 2376, 2378, 2380, 2382, 2384, 2386, 2388, 2390, 2392, 2394, 2396, 2398, 2400, 2402, 2404, 2406, 2408, 2410, 2412, 2414, 2416, 2418, 2420, 2422, 2424, 2426, 2428, 2430, 2432, 2434, 2436, 2438, 2440, 2442, 2444, 2446, 2448, 2450, 2452, 2454, 2456, 2458, 2460, 2462, 2464, 2466, 2468, 2470, 2472, 2474, 2476, 2478, 2480, 2482, 2484, 2486, 2488, 2490, 2492, 2494, 2496, 2498, 2500, 2502, 2504, 2506, 2508, 2510, 2512, 2514, 2516, 2518, 2520, 2522, 2524, 2526, 2528, 2530, 2532, 2534, 2536, 2538, 2540, 2542, 2544, 2546, 2548, 2550, 2552, 2554, 2556, 2558, 2560, 2562, 2564, 2566, 2568, 2570, 2572, 2574, 2576, 2578, 2580, 2582, 2584, 2586, 2588, 2590, 2592, 2594, 2596, 2598, 2600, 2602, 2604, 2606, 2608, 2610, 2612, 2614, 2616, 2618, 2620, 2622, 2624, 2626, 2628, 2630, 2632, 2634, 2636, 2638, 2640, 2642, 2644, 2646, 2648, 2650, 2652, 2654, 2656, 2658, 2660, 2662, 2664, 2666, 2668, 2670, 2672, 2674, 2676, 2678, 2680, 2682, 2684, 2686, 2688, 2690, 2692, 2694, 2696, 2698, 2700, 2702, 2704, 2706, 2708, 2710, 2712, 2714, 2716, 2718, 2720, 2722, 2724, 2726, 2728, 2730, 2732, 2734, 2736, 2738, 2740, 2742, 2744, 2746, 2748, 2750, 2752, 2754, 2756, 2758, 2760, 2762, 2764, 2766, 2768, 2770, 2772, 2774, 2776, 2778, 2780, 2782, 2784, 2786, 2788, 2790, 2792, 2794, 2796, 2798, 2800, 2802, 2804, 2806, 2808, 2810, 2812, 2814, 2816, 2818, 2820, 2822, 2824, 2826, 2828, 2830, 2832, 2834, 2836, 2838, 2840, 2842, 2844, 2846, 2848, 2850, 2852, 2854, 2856, 2858, 2860, 2862, 2864, 2866, 2868, 2870, 2872, 2874, 2876, 2878, 2880, 2882, 2884, 2886, 2888, 2890, 2892, 2894, 2896, 2898, 2900, 2902, 2904, 2906, 2908, 2910, 2912, 2914, 2916, 2918, 2920, 2922, 2924, 2926, 2928, 2930, 2932, 2934, 2936, 2938, 2940, 2942, 2944, 2946, 2948, 2950, 2952, 2954, 2956, 2958, 2960, 2962, 2964, 2966, 2968, 2970, 2972, 2974, 2976, 2978, 2980, 2982, 2984, 2986, 2988, 2990, 2992, 2994, 2996, 2998, 3000]

Task7 *:*

You need to write a code which accepts a sequence of comma separated 4 digit binary numbers as its input and then check whether they are divisible by 5 or not. The numbers that are divisible by 5 are to be printed in a comma separated sequence.

Example: 0100,001 1,1010, 1001 Then the output should be: 1010

```
In [93]: def divisible_5 ():
binary_nums = (input("Enter 4 digits binary numbers seperated by commas: "))
binary_nums
binary_nums.split(",")
divisible_nums = []
for num in binary_nums:
num = int(num)
if num % 5 == 0:
divisible_nums.append(num)
print(divisible_nums)

divisible_5()
```

```
Enter 4 digits binary numbers seperated by commas: 0100,0011, 1010, 1001

[100, 1010]
```

Task8:

Write a program that accepts a sentence and calculate the number of letters and digits.

Suppose the following input is supplied to the program: hello world! 123 Then, the output should be: LETTERS 10 DIGITS 3

```
In [102]: def check_letters_digits ():
```

```
In [ ]:
sentence = input("Enter a sentence with digits:")
digits=0
letters = 0
for char in sentence:
if char.isalpha():
letters += 1
elif char.isdigit():
digits += 1
print (f"""Letters: {letters}
Digits
: {digits}""")
check_letters_digits()
```

```
Enter a sentence with digits: hello world 123
Letters: 10
Digits
: 3
```