Python

1. Write a function that takes a list and returns the sum of all elements in the list(so not use the built in sum function)
2. Write a function that returns whether the string passed is palindrome or not(return true or false).
3. Write a function that returns whether the integer passed is palindrome or not(return true or false).
4. WAF to return whether the argument passed to it is palindrome or not. Code should work for both strings and integers. [**Hint**: Use the **isinstance** method]
5. WAF that takes a string and returns the number of upper and lower case characters in a single tuple.
6. What does the following do:

   def a(b, c, d): pass
7. Read help for zip and enumerate functions and print output of:

```
l1 = [1,2,3,4]
l2 = [4,3,2,1]
for t in zip(l1,l2):
    print(t[0], t[1])
```

```
l1 = [1,2,3]
l2 = [4,3,2,1]
for t in zip(l1,l2):
    print(t[0], t[1])
```

```
l1 = [1,2,3,4]
l2 = [4,3]
for t in zip(l1,l2):
    print(t[0]**t[1])
```

```
l1 = range(10,21,2)
for item in enumerate(l1):
    print(item[0], item[1], sep = '-')
```

8. Find output of following:

```
functs = [min, max, sum, len]
nums = [6, 7, 4, 3, 2]
for funct in functs:
    print(funct(nums))
```

```
functs = [min, max, sum, len]
nums = [6, 7, 4, 3, 2]
l = [funct(nums) for funct in functs]
```

```
functs = [min, max, sum, len]
nums = [6, 7, 4, 3, 2]
l = [funct(nums) for funct in functs]
for funct in functs:
    print(funct(l))
```

```
functs = [min, max, sum, len]
nums = [6, 7, 4, 3, 2]
l = [funct([funct(nums) for funct in functs]) for funct in functs]
print(l)
```

9. Write a function *is_keyword(word)* that returns whether passed argument is a C++ keyword or not. [Yes C++ keywords: Google the list of keywords available in C++ 17]
10. WAF to return whether the passed argument is a valid Python identifier or not.
11. Write a function hor_or_not() that prints whether the climate is hot or not:

Temp in C	**Result**
<0	Hot!! Crazy??
0 <= C < 10	Not Hot at all
10 <= C < 20	Not Hot
20 <= C < 30	Good time to Code
30 <= C < 40	Yes its Hot
40 <= C < 50	Yes its too Hot
C > 50	Still Alive ??

Update the function to allow the function to take temperature in both ‘C’ and ‘F’.
So the function can be called something like this:
hot_or_not(40 ,’C’)
hot_or_not(400 ,’F’)