Industrial Software Development (ISDe) – February 15, 2021

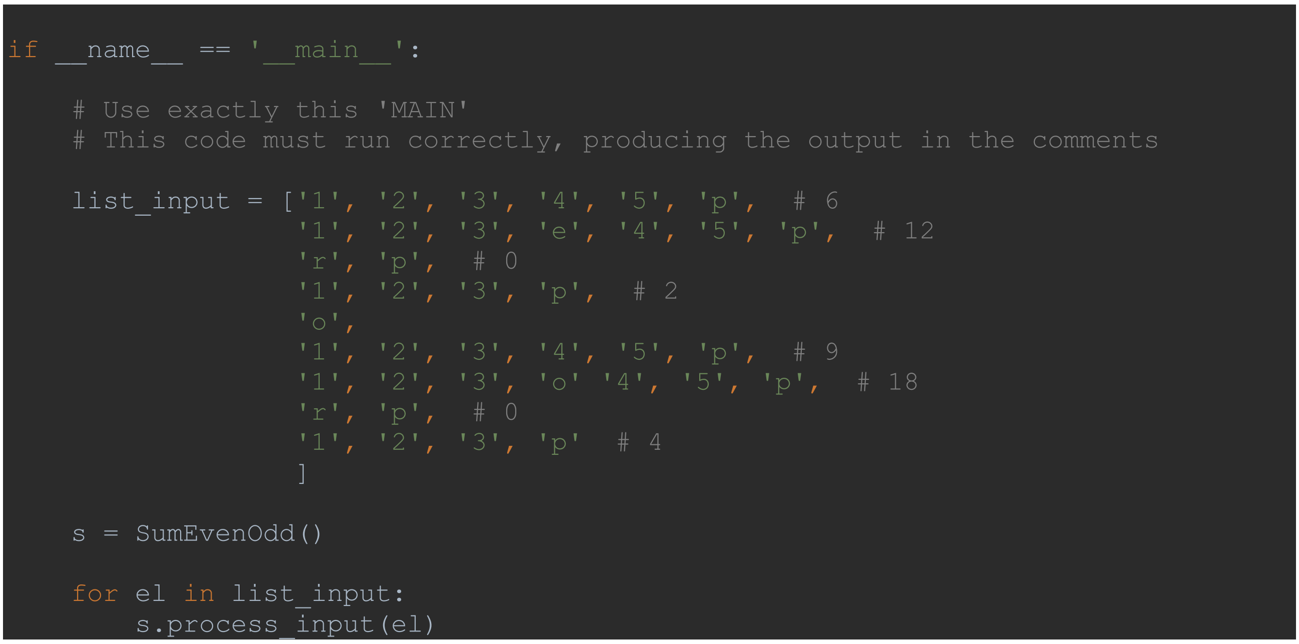
Name: .... Surname: .... Student ID: ....

EXERCISE 3 (15 points) - 40 minutes

Implement a state machine that receives one character at a time. The machine will add only even numbers or only odd numbers according to the following rules.

- Initially, the machine adds the even numbers.
- If it receives the '**o**' character, it goes to the state in which it adds the odd numbers, and the previous sum is deleted.
- If the character '**e**' is input, it goes to the state in which it adds the even numbers, and the previous sum is deleted.
- If it receives the '**o**' character when it is in the state in which it adds the odd numbers or receives the '**e**' character when it is in the state in which it adds the even numbers, the input has no effect
- If it receives the '**r**' character, it resets the sum
- If it receives the '**p**' character, it prints the sum.

Use the **state desing pattern** to implement this behavior.
Write your code into a single '**main_3.py**' file



```
if __name__ == '__main__':

    # Use exactly this 'MAIN'
    # This code must run correctly, producing the output in the comments

    list_input = ['1', '2', '3', '4', '5', 'p',   # 6
                  '1', '2', '3', 'e', '4', '5', 'p',   # 12
                  'r', 'p',   # 0
                  '1', '2', '3', 'p',   # 2
                  'o',
                  '1', '2', '3', '4', '5', 'p',   # 9
                  '1', '2', '3', 'o' '4', '5', 'p',   # 18
                  'r', 'p',   # 0
                  '1', '2', '3', 'p'   # 4
                  ]

    s = SumEvenOdd()

    for el in list_input:
        s.process_input(el)
```

The output is:

```
6
12
0
2
9
18
0
4
```