

```

1  class Observe:
2      """
3      The Observe class stores two values:
4      self.value      is the target value on a certain state variable of the observed system
5      self.variable    is the actual (probably changing) value of that state variable
6
7      self.value and self.variable can be set at instance initialization, or with the self.update() method
8      Additionally, there exist some methods that allow for easy comparison of self.variable vs self.value:
9
10         Method      |  .equ()  .neq()  .gtr()  .geq()  .lss()  .leq()  |
11         Equivalent   |  ==      !=      >      >=     <      <=     |
12         With these methods self.value will not be updated
13         """
14
15     def __init__(self,*args):
16         if len(args)==0:
17             self.initial=None
18             self.value=None
19             self.variable=None
20         elif len(args)==1:
21             self.initial=args[0]
22             self.value=None
23             self.variable=None
24         elif len(args)==2:
25             self.initial=args[0]
26             self.value=args[1]
27             self.variable=None
28         elif len(args)==3:
29             self.initial=args[0]
30             self.value=args[1]
31             self.variable=args[2]
32
33     def update(self,var,*args):
34         self.variable=var
35         if len(args)>0:
36             self.value=args[0]
37
38     def equ(self,*args):
39         if len(args)==1:      # If argument inserted, compare against it instead of using self.value
40             if self.variable==args[0]:
41                 return True
42             else:
43                 return False
44         else:
45             if self.variable==self.value:
46                 return True
47             else:
48                 return False
49
50     def neq(self,*args):
51         if len(args)==1:      # If argument inserted, compare against it instead of using self.value
52             if self.variable!=args[0]:
53                 return True
54             else:
55                 return False
56         else:
57             if self.variable!=self.value:
58                 return True
59             else:
60                 return False

```

```
58         return False
59     def gtr(self,*args):
60         if len(args)==1:      # If argument inserted, compare against it instead of u
sing self.value
61             if self.variable>args[0]:
62                 return True
63             else:
64                 return False
65         else:
66             if self.variable>self.value:
67                 return True
68             else:
69                 return False
70     def geq(self,*args):
71         if len(args)==1:      # If argument inserted, compare against it instead of u
sing self.value
72             if self.variable>=args[0]:
73                 return True
74             else:
75                 return False
76         else:
77             if self.variable>=self.value:
78                 return True
79             else:
80                 return False
81     def lss(self,*args):
82         if len(args)==1:      # If argument inserted, compare against it instead of u
sing self.value
83             if self.variable<args[0]:
84                 return True
85             else:
86                 return False
87         else:
88             if self.variable<self.value:
89                 return True
90             else:
91                 return False
92     def leq(self,*args):
93         if len(args)==1:      # If argument inserted, compare against it instead of u
sing self.value
94             if self.variable<=args[0]:
95                 return True
96             else:
97                 return False
98         else:
99             if self.variable<=self.value:
100                 return True
101             else:
102                 return False
```