

```

1 # Client Application (client.py)
2 # Bach Vu
3 # 01/08/2020
4
5 from request import *
6 from response import *
7 from socket import *
8 import sys, select
9
10 class DTClient():
11     def __init__(self, hostname, port):
12         self.socket = socket(AF_INET, SOCK_DGRAM)
13         self.socket.setblocking(0)
14         self.target = None
15         try:
16             hostname = getaddrinfo(hostname, port)[0][4][0]
17             self.target = (hostname, port)
18         except Exception as e:
19             raise e
20
21     def postRequest(self, request):
22         packet = request.encodePacket()
23         if isinstance(packet, bytearray):
24             self.socket.sendto(packet, self.target)
25             print("Request sent to {}:{}! Waiting for response ... ".format(self.target[0],
self.target[1]))
26         else:
27             print("Request sent failed with code {}! Try again ... ".format(packet))
28
29     def getResponse(self):
30         ready = select.select([self.socket], [], [], 1)
31         if ready[0]:
32             data, addr = self.socket.recvfrom(1024)
33             return data
34         return None
35
36 ##### Main Program #####
37 def main():
38     print("\nWelcome to DT Finder (Client)")
39     # Error Checking
40     errMess = [
41         "Argument input error.\n    python client.py {mode} {host_target}
{port_eng/maori/ger}",
42         "Ports input must be integer (whole number).",
43         "Port must be between 1024 and 64000 inclusively!",
44         "mode must be 'time' or 'date'."
45     ]
46     errCode = checkInputArgv()
47     if errCode != 0:
48         print(errMess[errCode-1])
49         sys.exit()
50
51     # Request
52     host, port = sys.argv[2], int(sys.argv[3])
53     DT_client = DTClient(host, port)
54     mo = 1 if sys.argv[1] == 'date' else 2
55     request = DT_Request(mo)
56     DT_client.postRequest(request)
57
58     # Response
59     response = DT_client.getResponse()

```

```
60     if not response: # None type
61         print("We received no data after 1sec (time-out).\n")
62         return
63     response = DT_Response.decodePacket(response, mo)
64
65     if isinstance(response, int):
66         err = "A response discarded with error Code {}:{}\n{}\n"
67         err_mess = DT_Response.ErrorMessage[response-1]
68         print(err.format(response, err_mess))
69     elif response:
70         print(response)
71         print()
72
73 def checkInputArgv():
74     if len(sys.argv) != 4:
75         return 1
76
77     mode = sys.argv[1]
78     try:
79         port = int(sys.argv[3])
80     except BaseException:
81         return 2
82
83     if port < 1024 or port > 64000:
84         return 3
85     if mode != "date" and mode != "time":
86         return 4
87     return 0
88
89 if __name__ == "__main__":
90     try:
91         main()
92     except Exception as e:
93         print(e)
94         print("Program exited unexpectedly.\n")
95
```