Assumptions and Description.

1. This is a basic simulation of the model describing how the vaccines are stored.

---

2. I have studied about the vaccine pfizer as I found the model described in the task pretty similar to the way this vaccine is stored.

---

3. Furthermore, I have decided to keep the slope of the graph to be 3K(3 Kelvin) per 5 hours which would make the total duration for the most of the simulations more than atleat 10 days which is the average span of pfizer.

---

4. I have taken random time instances for replenishing the cryogenic liquid, as 1/15 so as to reduce the cost factor.

---

5. The Total duration for the vaccine not to be unfit is not going to be fixed how ever it would surely be more than 9-10 days and less than 15 days.

```
from random import randint
To = 77
slope = 0.60
t = 0
T = 0
tempvstime = [77]
counter = 0
while T <= 200:
  t = t + 20
  T = (slope)*t + To
  value = randint(0, 15)
  if value == 1:
    T = 77
    t = 0
  tempvstime.append(T)
  counter = counter + 1
  if counter == 18:
    break
print("The simulation ends as time reaches ", counter*20 , "hours", "that is nearly ", (counter*20)//24 , "days")
import pandas as pd
h = []
for i in range(len(tempvstime)):
  h.append([i*20 , tempvstime[i]])
pd.DataFrame(h, columns=["-Time in hours-", "-TEMPRATURE-"])
```

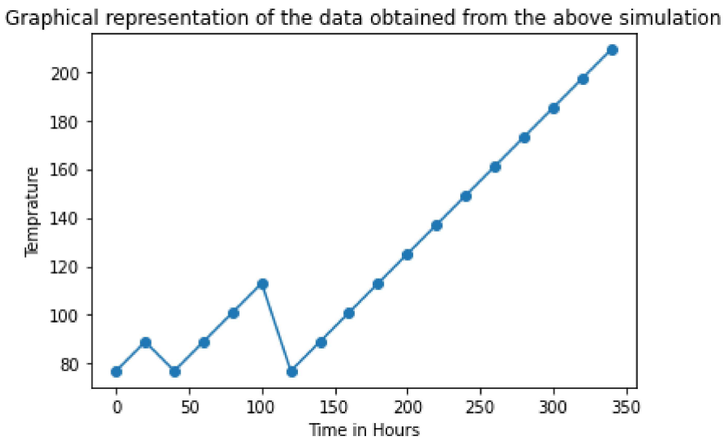
```
The simulation ends as time reaches  340 hours that is nearly  14 days
```

	-Time in hours-	-TEMPRATURE-
0	0	77.0
1	20	89.0
2	40	77.0
3	60	89.0
4	80	101.0
5	100	113.0
6	120	77.0
7	140	89.0
8	160	101.0
9	180	113.0
10	200	125.0
11	220	137.0
12	240	149.0
13	260	161.0
14	280	173.0

Ploting The Thermocouple Data

```
import matplotlib.pyplot as plt
x = [0, 20, 40, 60, 80, 100, 120, 140, 160, 180, 200, 220, 240, 260, 280, 300, 320, 340]
y = [77.0,  89.0, 77.0, 89.0, 101.0,  113.0,  77.0, 89.0, 101.0, 113.0, 125.0,  137.0,  149.0,  161.0,  173.0,  185.0,  197.0,  209.0]
plt.xlabel('Time in Hours')
plt.ylabel('Temprature')
plt.plot(x, y)
plt.title("Graphical representation of the data obtained from the above simulation")
plt.scatter(x,y)
```

```
<matplotlib.collections.PathCollection at 0x7f704ba69110>
```



✓ 0s completed at 11:47 AM