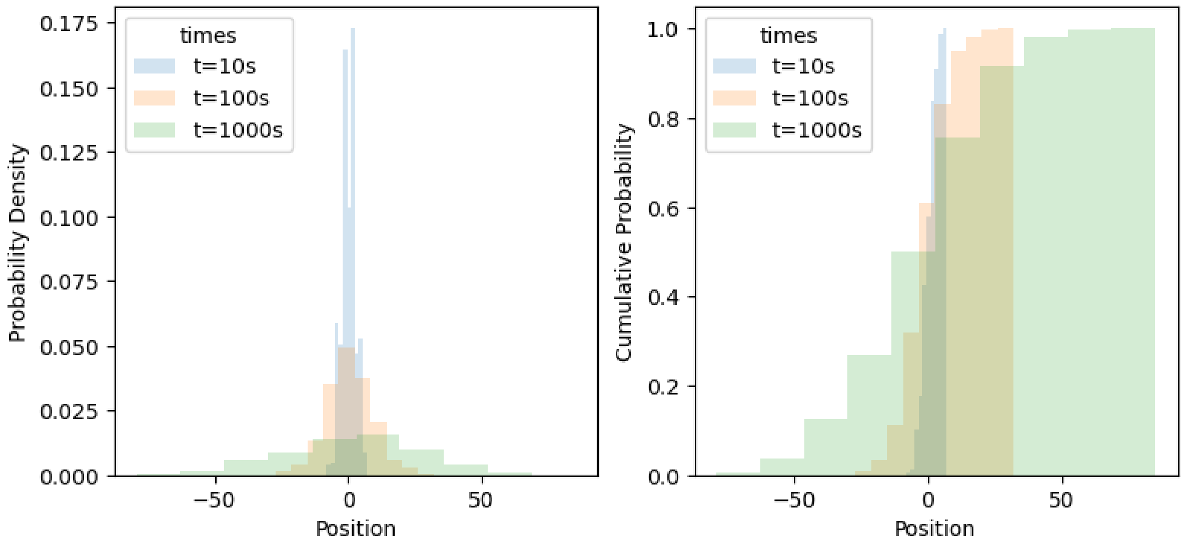
In [26]:

```
#Import modules
import numpy as np
np.random.seed()
import matplotlib.pyplot as plt
```

In [27]:

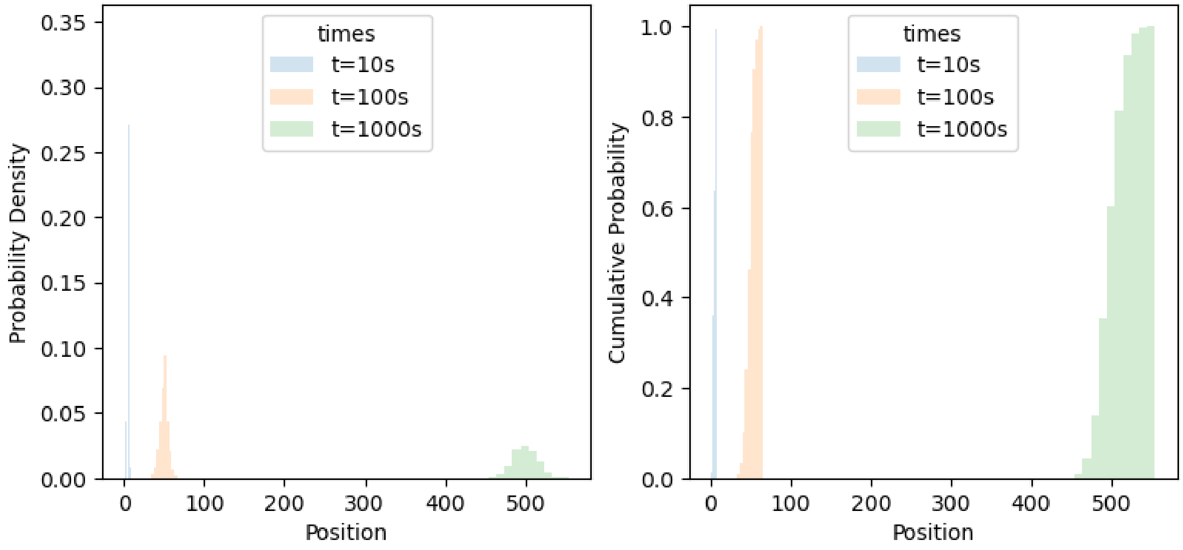
```
Nruns=1000 #simulate random walks many times
Nsteps=1000 #time steps

#initialize variables for t=10, t= 100 and t=1000 positions of each walk
times=np.zeros([3,Nruns],int)

#loop over walks
for j in range(Nruns):
    position=np.zeros(Nsteps) # variable to store every time steps' position
    position[0]=0 #walk starts at x=0

    #make a dice for 3 options with probabilities of each condition =1/3
    dice = np.random.choice(3,Nsteps,p=[1/3,1/3,1/3])

    for i in range(Nsteps):
        if dice[i]== 0: #Pl
            position[i]=position[i-1]-1
        elif dice[i]== 1: #Pr
            position[i]=position[i-1]+1
        else:
            position[i]=position[i-1] #Ps

    times[0][j]=position[9] #t=10, postion at (j+1)-th walk
    times[1][j]=position[99] #t=100
    times[2][j]=position[999] #t=1000

# set subplots and figure size
fig, (ax1, ax2)=plt.subplots(1,2, figsize=(9,4))
labels=['t=10s','t=100s','t=1000s'] #set labels

#subplot1: PDF for three time points
for k in range(3):
    ax1.hist(times[k], bins=10, density=True, alpha=0.2,cumulative=False,lab

ax1.set_ylabel('Probability Density')
ax1.set_xlabel('Position')
ax1.legend(loc='upper left',title='times')

#subplot2:CDF for three different time points
for k in range(3):
    ax2.hist(times[k], bins=10, density=True, alpha=0.2,cumulative=True,labe

ax2.set_ylabel('Cumulative Probability')
ax2.set_xlabel('Position')
ax2.legend(loc='upper left',title='times')

plt.show()
```



In [29]:

```
#same two plots for pl = 0, pr = 1/2, and ps = 1/2

Nruns=1000 #simulate random walks many times
Nsteps=1000 #time steps

#initialize variables for t=10, t= 100 and t=1000 positions of each walk
times=np.zeros([3,Nruns],int)

#loop over walks
for j in range(Nruns):
    position=np.zeros(Nsteps) # variable to store every time steps' position
    position[0]=0.0 #walk starts at x=0
    #make a dice for 3 options with probabilities of each condition
    dice = np.random.choice(3,Nsteps,p=[0,1/2,1/2])
    #loop through steps
    for i in range(Nsteps):
        if dice[i]==0: #Pl
            position[i]=position[i-1]-1
        elif dice[i]==1: #Pr
            position[i]=position[i-1]+1
        elif dice[i]==2: #Ps
            position[i]=position[i-1]
    times[0][j]=position[9]
    times[1][j]=position[99]
    times[2][j]=position[Nsteps-1]

#set subplots and figure size
fig, (ax1, ax2)=plt.subplots(1,2, figsize=(9,4))
labels=['t=10s','t=100s','t=1000s'] #set labels

#subplot1: PDF for three time points
for k in range(3):
    ax1.hist(times[k], bins=10, density=True, alpha=0.2,cumulative=False,lab

ax1.set_ylabel('Probability Density')
ax1.set_xlabel('Position')
ax1.legend(loc='upper center',title='times')

#subplot2:CDF for three different time points
for k in range(3):
    ax2.hist(times[k], bins=10, density=True, alpha=0.2,cumulative=True,labe

ax2.set_ylabel('Cumulative Probability')
ax2.set_xlabel('Position')
ax2.legend(loc='upper center',title='times')
```

```
plt.show()
```


Probability Density
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
0
100
200
300
400
500
Position
times
t=10s
t=100s
t=1000s
Cumulative Probability
1.0
0.8
0.6
0.4
0.2
0.0
0
100
200
300
400
500
Position
times
t=10s
t=100s
t=1000s

In [ ]: