Synapses

Definition		
Basic	`S = Synapses(P, Q, model = 'w : 1', pre = 'v += w')`	

Stochastic	`S=Synapses(input, neurons,`
	`  model='''w : 1`
	`  p : 1''', pre="v+=w*(rand()<p)")`	

Nonlinear	`S=Synapses(input, neurons,`
	`  model='''dg/dt=-a*g+b*x*(1-g) : 1`
	`  dx/dt=-c*x : 1`
	`  w : 1''', pre='x+=w')`
	`neurons.gtot=S.g # gtot receives sum over presynaptic neurons`	

Complex updates	`S=Synapses(input,neuron,`
	`  model='''x : 1`
	`  u : 1`
	`  w : 1''',`
	`  pre='''u=U+(u-U)*exp(-(t-lastupdate)/tauf)`
	`  x=1+(x-1)*exp(-(t-lastupdate)/taud)`
	`  i+=w*u*x`
	`  x*=(1-u)`
	`  u+=U*(1-u)''')`	

Event-driven	`S=Synapses(input, neurons,`
	`  model='''w:1`
	`  dApre/dt=-Apre/tpre : 1 (event-driven)`
	`  dApost/dt=-Apost/tpost : 1 (event-driven)''',`
	`  pre='''ge+=w`
	`  Apre+=dApre`
	`  w=clip(w+Apost,0,gmax)''',`
	`  post='''Apost+=dApost`
	`  w=clip(w+Apre,0,gmax)''')`	

Gap junctions	`S=Synapses(neurons,model='''w:1`
	`  Igap=w*(v_pre-v_post): 1''')`
	`neurons.Igap=S.Igap`	

Construction		
Single synapse	`S[0,2]=True`	
Random	`S[:,:]=0.02`	
Condition	`S[:,:]='j==((i+1)%N)'`	
Subgroups	`S[Pe,Q]=True`	
Multiple synapses	`S[0,0]=2 # creates 2 synapses`	
Assignment		
Single synapse	`S.w[0,2]=1*mV`	
Multiple synapses	`S.w[0,0,0]=1*mV` `S.w[0,0,1]=.5*mV`	
Vectorized	`S.w[0,:]=1*mV`	
Expression	`S.w='rand()*wmax'` `S.w='.4*mV*cos(2*pi*(i-j)*1./N)'`	

Delays	`S.delay[0,2]=2*ms`
	`S.delay='rand()*2*ms'`
	`S.delay[0,0,0]=.5*ms`	

Monitoring		
	`M=StateMonitor(S,'w',record=[0,1])`	Monitors synapses number 0 and 1