

## The Cluster Visualize Function

```
1 def visualize(self, i):
2     """
3     Shows a snapshot of the current state of the
4     system at time t_i. Particles are scaled by
5     size.
6     """
7     fig = plt.figure()
8     ax = fig.add_subplot(111, projection='3d')
9     ax.set_xlim3d([-50, 50])
10    ax.set_ylim3d([-50, 50])
11    ax.set_zlim3d([-50, 50])
12    ax.view_init(elev=11, azim=360*float(i)/self.
13                n_steps)
14    for body in self.cb_list:
15        ax.scatter(body.r[i,0], body.r[i,1], body.r[i
16                    ,2], s=body.radius, c=body.radius
17                    **4/70**4, cmap='autumn')
18    plt.axis('off')
19    plt.savefig('tmp_%04d.png' % i)
20    plt.close()
```

Listing 1: Cluster.py

## Does it do C++?

```
1 void main(int argc, char* argv[]) {
2     std::cout << "This is a test" << std::endl;
3     return 0;
4 }
```

Listing 2: Testing.cpp