```
from visual import*
from random import random
print("The size of space is one square meter")##output introductory words
num=500##input number
r=0.03##input the radius of balls
v=0.2##input the velocity of balls
wallR=box(pos=vector(0.5,0,0),size=vector(0.02,1,1),color=color.white)
wallL=box(pos=vector(-0.5,0,0),size=vector(0.02,1,1),color=color.white)
wallA=box(pos=vector(0,0,-0.5),size=vector(1,1,0.02),color=color.white)
wallB=box(pos=vector(0,0,0.5),size=vector(1,1,0.02),color=color.white,opacity=0.1)
wallU=box(pos=vector(0,0.5,0),size=vector(1,0.02,1),color=color.white)
wallD=box(pos=vector(0,-0.5,0),size=vector(1,0.02,1),color=color.white)##define walls
def ball(x,y):
    ball=sphere(pos=vector(x,y,0),color=vector(1,random(),random()),radius=r)
    X=random()
    Y=pow((1-X*X),0.5)
    ball.velocity=v*vector(X,Y,0)
    return(ball)
box=[]
for i in range(num):
    bi=ball(random()-0.5,random()-0.5)
    box.append(bi)##define balls
def pz_wall(ball):
    if ball.pos.x>wallR.pos.x-0.02-r:
        ball.velocity.x=-ball.velocity.x
    if ball.pos.x<wallL.pos.x+0.02+r:
        ball.velocity.x=-ball.velocity.x
    if ball.pos.y<wallD.pos.y+0.02+r:
        ball.velocity.y=-ball.velocity.y
    if ball.pos.y>wallU.pos.y-0.02-r:
        ball.velocity.y=-ball.velocity.y##define collision between balls and wall(s)
t=0
dt=0.001
while True:
    rate(1000)
    for i in range(num):
        pz_wall(box[i])
        for j in range(num):
            pz_wall(box[j])
            if i!=j:
                box[i].pos+=box[i].velocity*dt
                box[j].pos+=box[j].velocity*dt
                x=abs(box[i].pos.x-box[j].pos.x)
                y=abs(box[i].pos.y-box[j].pos.y)
```

```
                    d=pow((x*x+y*y),0.5)
                    if d<r+r:
                        B=2*r*random()
                        A=r
                        a=(B*B-2*A*A)/(2*A*A)
                        b=1-a
                        r=B*pow((4*A*A-B*B),0.5)/(2*A*A)
                        m=0.5

v1x=m*(1+a)*box[i].velocity.x+m*b*box[j].velocity.x-m*r*box[i].velocity.y+m*r*box[j].velocity.y

v2x=m*b*box[i].velocity.x+m*(1+a)*box[j].velocity.x+m*r*box[i].velocity.y-m*r*box[j].velocity.y

v1y=m*r*box[i].velocity.x-m*r*box[j].velocity.x+m*(1+a)*box[i].velocity.y+m*b*box[j].velocity.y

v2y=-m*r*box[i].velocity.x+m*r*box[j].velocity.x+m*b*box[i].velocity.y+m*(1+a)*box[j].velocity.y
                        box[i].velocity.x=v1x
                        box[j].velocity.x=v2x
                        box[i].velocity.y=v1y
                        box[j].velocity.y=v2y
                        t=t+1
                        if t==2500:
                            for k in range(num):

v=pow(box[k].velocity.x*box[k].velocity.x+box[k].velocity.y*box[k].velocity.y,0.5)
                                    print v
```