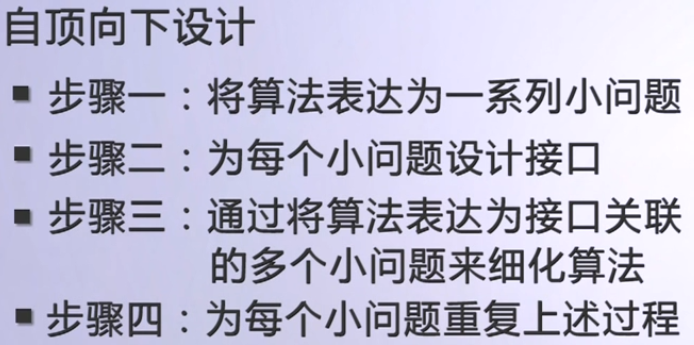
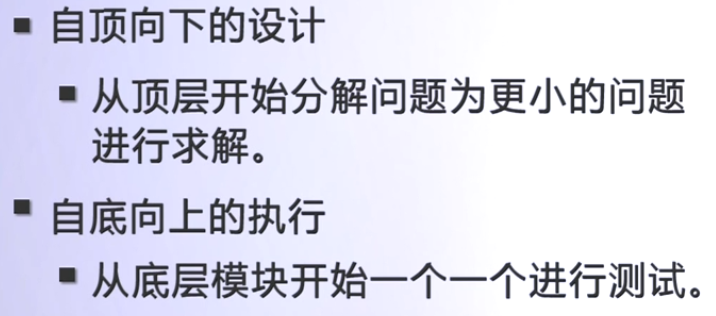
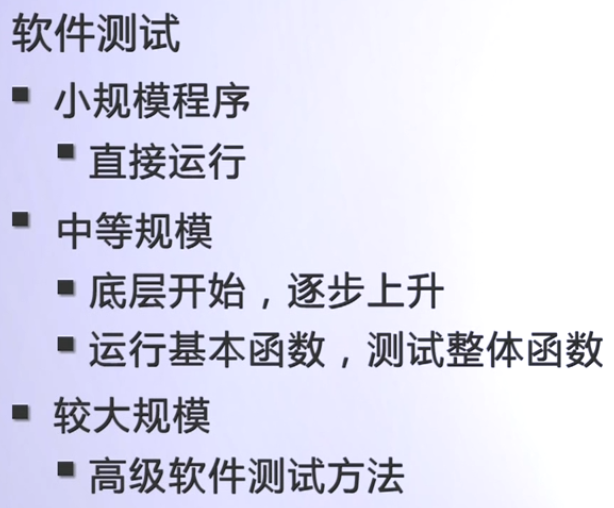
程序设计方法

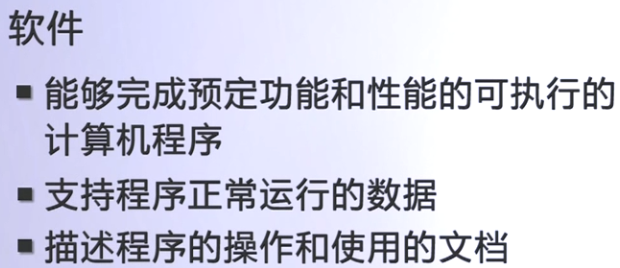
# 自顶向下设计

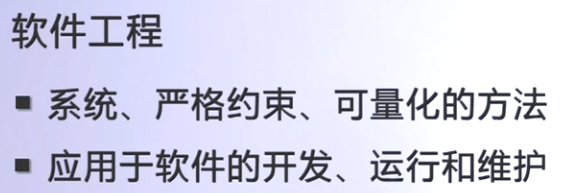




# 软件测试







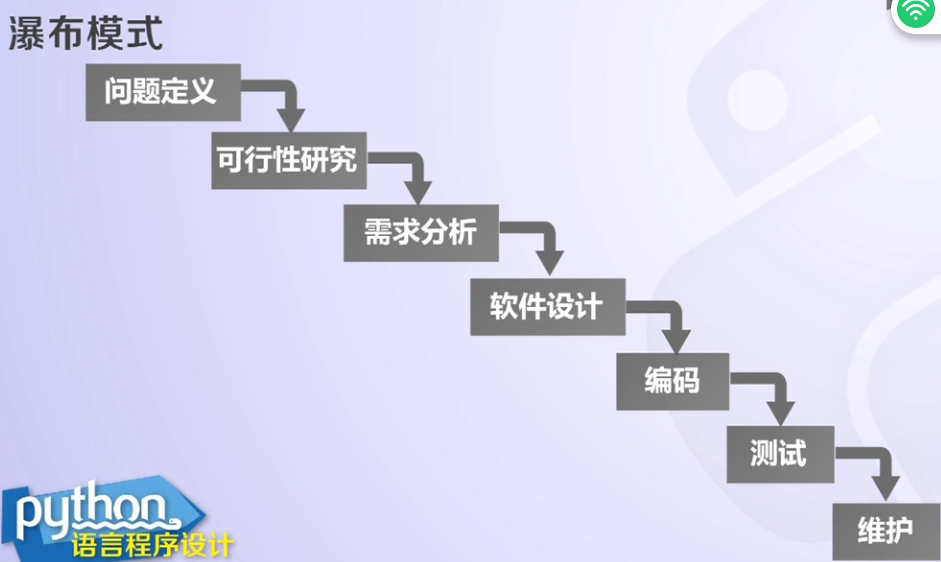
# 软件开发生命周期



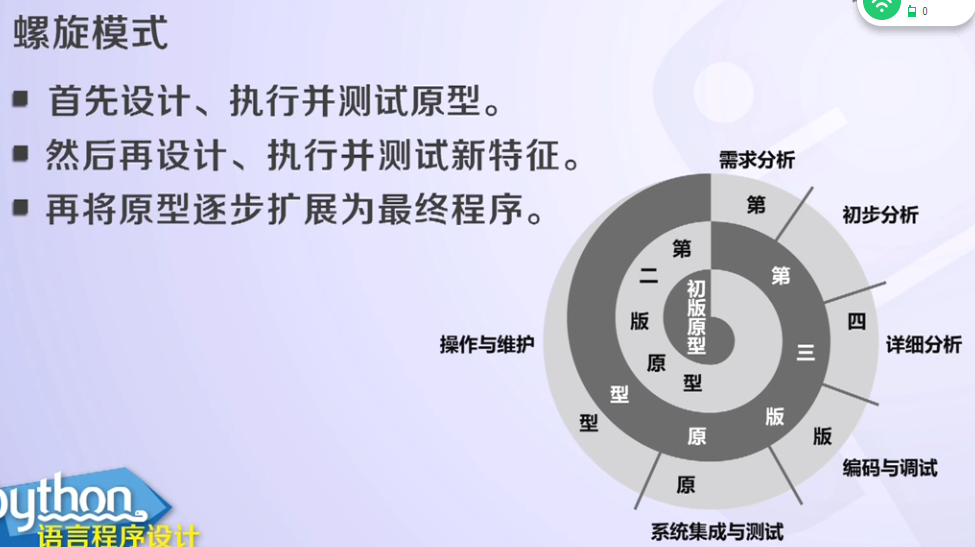
# 常见的软件开发模式



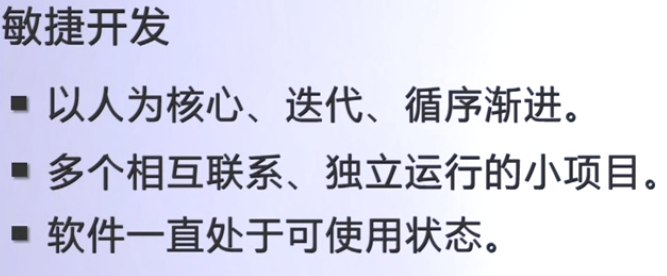
## 瀑布模式

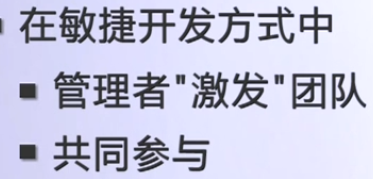


## 螺旋模式

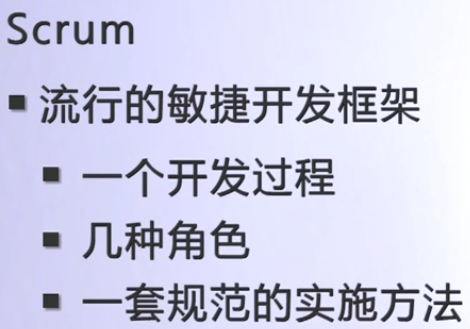


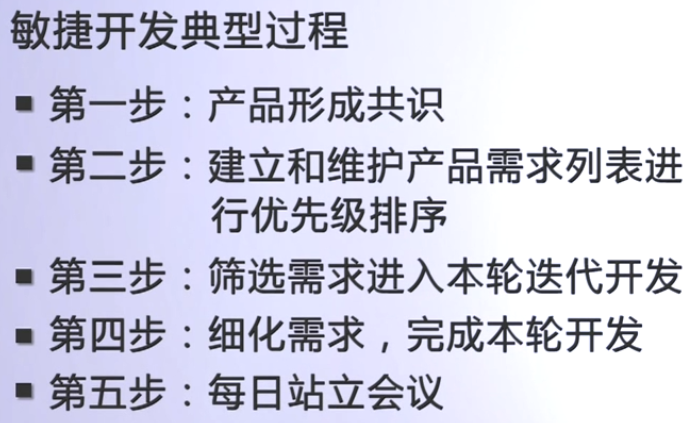
## 敏捷开发模式：

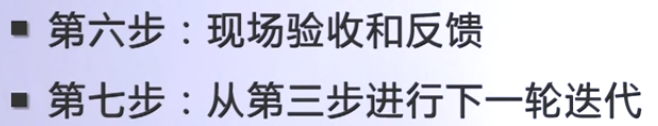




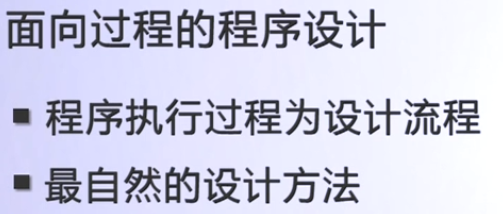




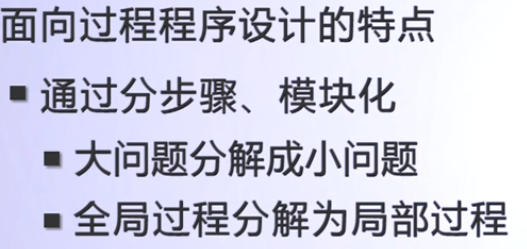


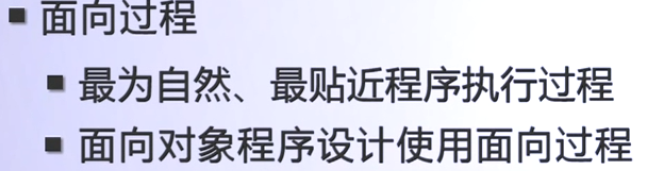


# 面向过程的程序设计（结构化编程）

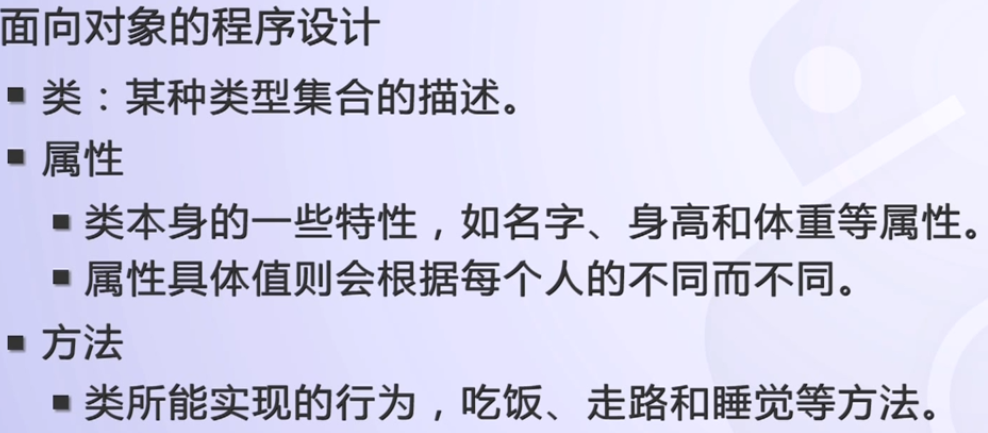


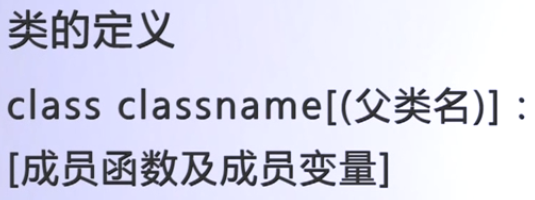


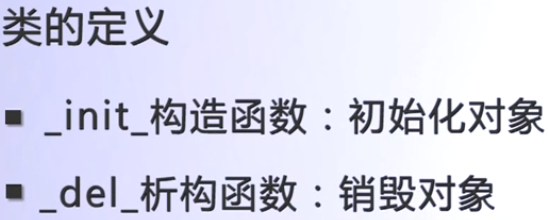


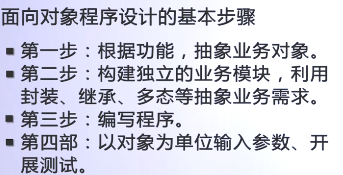


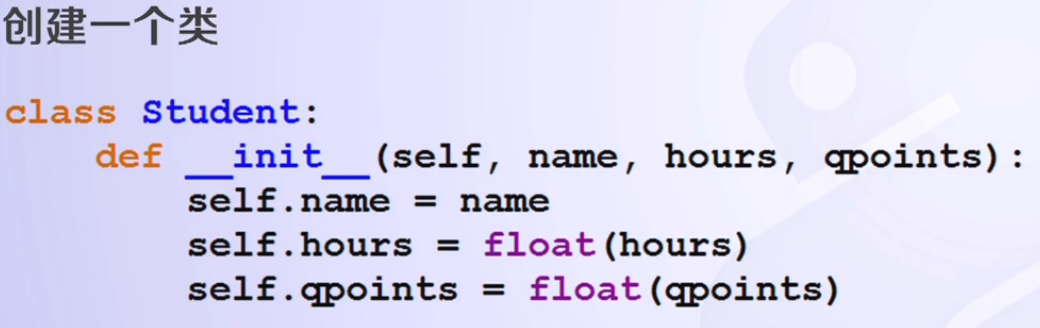
# 面向对象设计

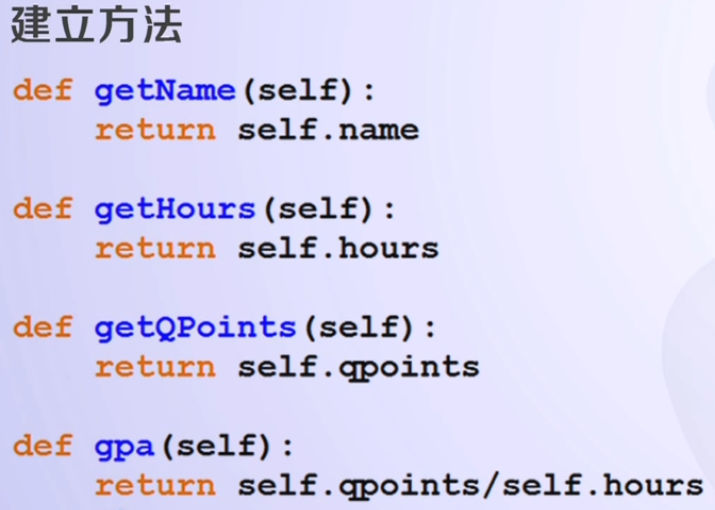


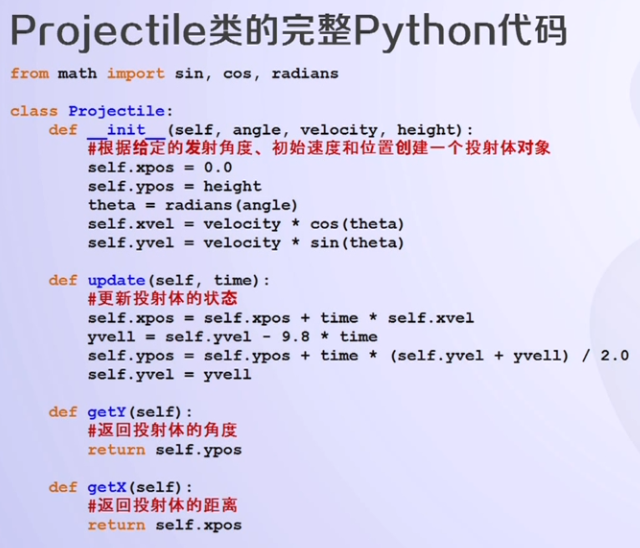


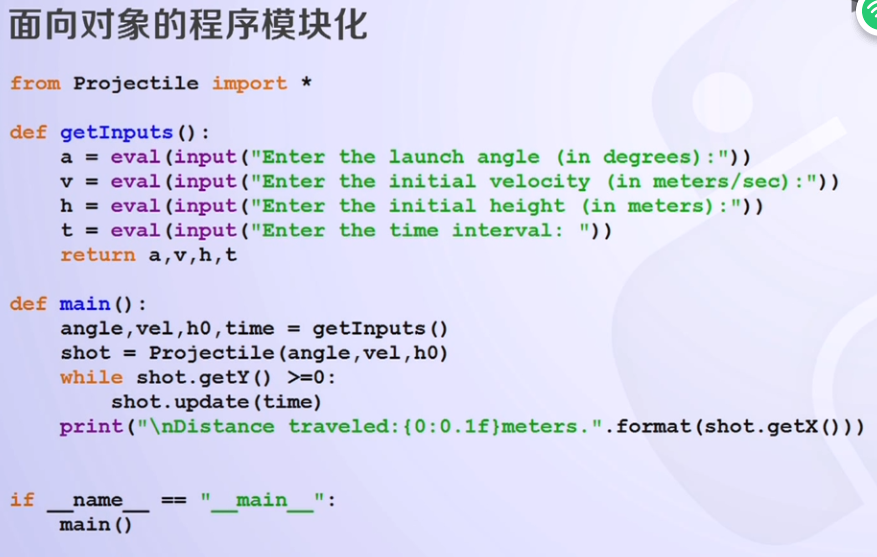




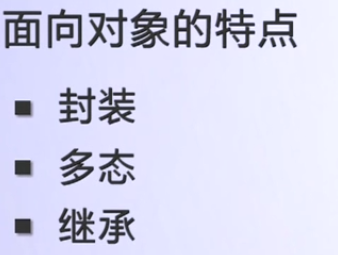


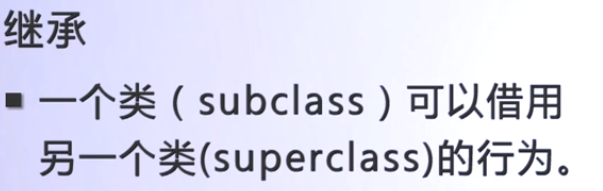






# 面向对象的特点





# 面向对象设计实例

## 面向过程设计：

from math import pi, sin, cos, radians  
  
def main():  
 angle = eval(input("Enter the launch angle (in degrees):"))  
 vel = eval(input("Enter the initial velocity (in meters/sec):"))  
 h0 = eval(input("Enter the initial height (in meters):"))  
 time = eval(input("Enter the time interval: "))  
 xpos = 0  
 ypos = h0  
 theta = radians(angle)  
 xvel = vel \* cos(theta)  
 yvel = vel \* sin(theta)  
  
 while ypos >= 0:  
 xpos = xpos + time \* xvel  
 yvell = yvel - time \* 9.8  
 ypos = ypos + time \* (yvel + yvell) / 2.0  
 yvel = yvell  
 print("\nDistance traveled:{0:0.1f}meters.".format(xpos))  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 main()

## 面向对象设计：

Projectile.py

from math import sin, cos, radians  
class Projectile:  
 def \_\_init\_\_(self, angle, velocity, height):  
 # 根据给定的发射角度、初始速度和位置创建一个投射体对象  
 self.xpos = 0.0  
 self.ypos = height  
 theta = radians(angle)  
 self.xvel = velocity \* cos(theta)  
 self.yvel = velocity \* sin(theta)  
 def update(self, time):  
 # 更新投射体的状态  
 self.xpos = self.xpos + time \* self.xvel  
 yvell = self.yvel - 9.8 \* time  
 self.ypos = self.ypos + time \* (self.yvel + yvell) / 2.0  
 self.yvel = yvell  
  
 def getY(self):  
 # 返回投射体的角度  
 return self.ypos  
  
 def getX(self):  
 # 返回投射体的距离  
 return self.xpos

shot2.py

from Projectile import \*  
  
def getInputs():  
 a = eval(input("Enter the launch angle (in degrees):"))  
 v = eval(input("Enter the initial velocity (in meters/sec):"))  
 h = eval(input("Enter the initial height (in meters):"))  
 t = eval(input("Enter the time interval: "))  
 return a, v, h, t  
def main():  
 angle, vel, h0, time = getInputs()  
 shot = Projectile(angle, vel, h0)  
 while shot.getY() >= 0:  
 shot.update(time)  
 print("\nDistance traveled:{0:0.1f}meters.".format(shot.getX()))  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 main()

