#### 面向对象程序设计的三大特点是什么

1封装;

2继承;

3多态;

#### 定义一个程序员类(coder) ，定义一个工作(work)方法，睡觉(sleep)方法, 一个幻想(imagine)方法, 一个存款(money)属性,其中幻想方法与存款属性类型为私有

|  |
| --- |
| **class** coder:  **def** \_\_init\_\_(self):  self.\_\_money = 10000   **def** work(self):  **pass   def** sleep(self):  **pass   def** \_\_imagine(self):  **pass** |

#### 定义一个小王类(xiaowang)继承自程序员类,并覆盖程序员类coder的work方法

|  |
| --- |
| **class** coder:  **def** work(self):  **pass  class** xiaowang(coder):  **def** work(self):  print(**"小王的work方法"**)  x = xiaowang() x.work() |

#### 小王类中添加一个类属性: 公司(company),值为”传智播客”

|  |
| --- |
| **class** xiaowang:  company = **"传智播客"  def** work(self):  print(**"小王的work方法"**) |

#### 小王类中添加一个类方法show\_company,可以显示类属性company的值

|  |
| --- |
| **class** xiaowang:  company = **"传智播客"  def** work(self):  print(**"小王的work方法"**)   @classmethod  **def** show\_company(cls):  print(cls.company)  xiaowang.show\_company() |

#### 小王类中添加一个静态方法test, 调用类的静态静态方法test, 显示”这是一个静态方法”

|  |
| --- |
| **class** xiaowang:  @staticmethod  **def test**():  print(**"这是一个静态方法"**)  xiaowang.now() |