## dict = {"k1":"v1","k2":"v2","k3":"v3"}

#1、请循环遍历出所有的key

#2、请循环遍历出所有的value

# 3、请在字典中增加一个键值对,"k4":"v4"

dict = {"k1":"v1","k2":"v2","k3":"v3"}  
for i in dict.keys():  
 print(i)  
for i in dict.values():  
 print(i)  
  
dict["k4"]="v4"  
print(dict)

## 小明去超市购买水果，账单如下

苹果 32.8

香蕉 22

葡萄 15.5

请将上面的数据存储到字典里，可以根据水果名称查询购买这个水果的费用

用水果名称做key，金额做value，创建一个字典

info = {

'苹果':32.8,

'香蕉': 22,

'葡萄': 15.5

}

小明，小刚去超市里购买水果

小明购买了苹果，草莓，香蕉，小刚购买了葡萄，橘子，樱桃，请从下面的描述的字典中，计算每个人花费的金额，并写入到money字段里。

以姓名做key，value仍然是字典

Friuts = {

‘苹果’：12.3， # 水果和单价

‘草莓’：4.5，

‘香蕉’：6.3，

‘葡萄’：5.8，

‘橘子’：6.4，

‘樱桃’：15.8

}

info = {

'小明': {

'fruits': {'苹果':4, '草莓':13, '香蕉':10},

'money': 170.7

},

'小刚': {

'fruits': {'葡萄':19, '橘子':12, '樱桃':30},

'money': 661

}

}

## 编写一个函数，传入一个列表，并统计每个数字出现的次数。返回字典数据：{21:3,56:9,10:3} （阿里一轮笔试题）

|  |
| --- |
| list1=[21,21,21,56,56,56,56,56,56,56,56,56,10,10,10] list2=list(set(list1)) t={} for i in list2:  t[i]=list1.count(i) print(t) |

## 有以下公司员工信息，将数据转换为字典方式（姓名作为键，其他作为值,张三:{xxx:xxx,xx:xxx}）

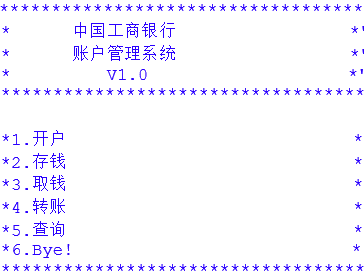
# 姓名 年龄 性别 编号 任职公司 薪资 部门编号

names = [  
 [**"刘备"**,**"56"**,**"男"**,**"106"**,**"IBM"**, 500 ,**"50"**],  
 [**"大乔"**,**"19"**,**"女"**,**"230"**,**"微软"**, 501 ,**"60"**],  
 [**"小乔"**, **"19"**, **"女"**, **"210"**, **"Oracle"**, 600, **"60"**],  
 [**"张飞"**, **"45"**, **"男"**, **"230"**, **"Tencent"**, 700 , **"10"**]  
]

names = [  
 ["刘备","56","男","106","IBM", 500 ,"50"],  
 ["大乔","19","女","230","微软", 501 ,"60"],  
 ["小乔", "19", "女", "210", "Oracle", 600, "60"],  
 ["张飞", "45", "男", "230", "Tencent", 700 , "10"]  
]  
d={}  
i=0  
for i in range(len(names)):  
  
 d[names[i][0]]={"姓名":names[i][0],"年龄":names[i][1],"性别":names[i][2],"编号":names[i][3],"工作部门":names[i][4],"薪资":names[i][5],"部门编号":names[i][6]}  
print(d)

## 各组，开始分析《中国工商银行账户管理系统》需求

1. 编程实现：仔细业务之间的包含关系，并完成以下编程需求，要适当提高代码的可复用性。
   1. **用户**：账号（str：系统随机产生8位数字）、姓名(str)、密码(int:6位数字)、地址、存款余额(int)、开户行（银行的名称（str））写死的！
   2. **地址**：国家(str)、省份(str)、街道(str)、门牌号(str)
   3. **银行**：能存储100用户的库(字典)、本银行名称（比如：中国工商银行的昌平支行,str）
      1. 银行业务功能
         1. 添加用户（传入参数：用户的所有信息。返回值：整型值（1：成功，2：用户已存在，3：用户库已满））
            1. 业务逻辑：

* 先检查该用户的账号在库里是否存在。若不存在则在用户库里添加一个该用户并返回代号1
* 若存在则返回代号2。另外在添加用户的时候检测用户库是否已注册满，若已满则返回代号3
  + - 1. 存钱（传入值：用户的账号、存取的金额。返回值：布尔类型值）
         1. 业务逻辑：
* 先根据传入的账号信息查询用户库里是否有该用户。若没有则返回False
* 若有，则将该用户的金额存进去。
  + - 1. 取钱（传入值：用户的账号，用户密码，取钱金额。返回值：整型值（0：正常，1：账号不存在，2：密码不对，3：钱不够））
         1. 业务逻辑：
* 先根据账号信息来查询该用户是否存在，若不存在，则返回代号1，
* 若存在，则继续判断密码是否正确，若不正确，则返回代号2。
* 若账号密码都正确，则继续判断当前用户的金额是否满足要取出的钱，若不满足，则返回代号3，
* 若满足，则将该用户的金额减去。
  + - 1. 转账（传入值：转出的账号，转入的账号，转出账号的密码，转出的金额。返回值：0：正常，1：账号不对，2密码不对，3钱不够）
         1. 业务逻辑：
* 先查询用户库是否存在转出和转入的账号，若不存在则返回代号,1，
* 若账号都存在则继续判断转出账号的密码是否正确，若不正确，则返回2，
* 若正确则继续判断要转出的金额是否足够，若不够则返回3，
* 否则正常转出，转出的账号用户金额要相对应的减少，转入的金额相对应的增加。
  + - 1. 查询账户功能（传入值：账号，账号密码，返回值：空）
         1. 业务逻辑：
* 先根据账号判断用户库是否存在该用户，若不存在则打印提示信息：该用户不存在。
* 否则继续判断密码是否正确。若不正确则打印相对应的错误信息。
* 若账号和密码都正确，则将该用户的信息都打印出来，比如：当前账号：xxxx,密码:xxxxxx,余额：xxxx元，用户居住地址：xxxxxxxxxxxxx，当前账户的开户行：xxxxxxxxxx.
  1. **界面类**：在执行该入口程序时，就打印银行业务选择菜单：比如：
     1. 
     2. 然后就开始处理各种输入操作，直到业务处理完成!