

神秘的宝藏

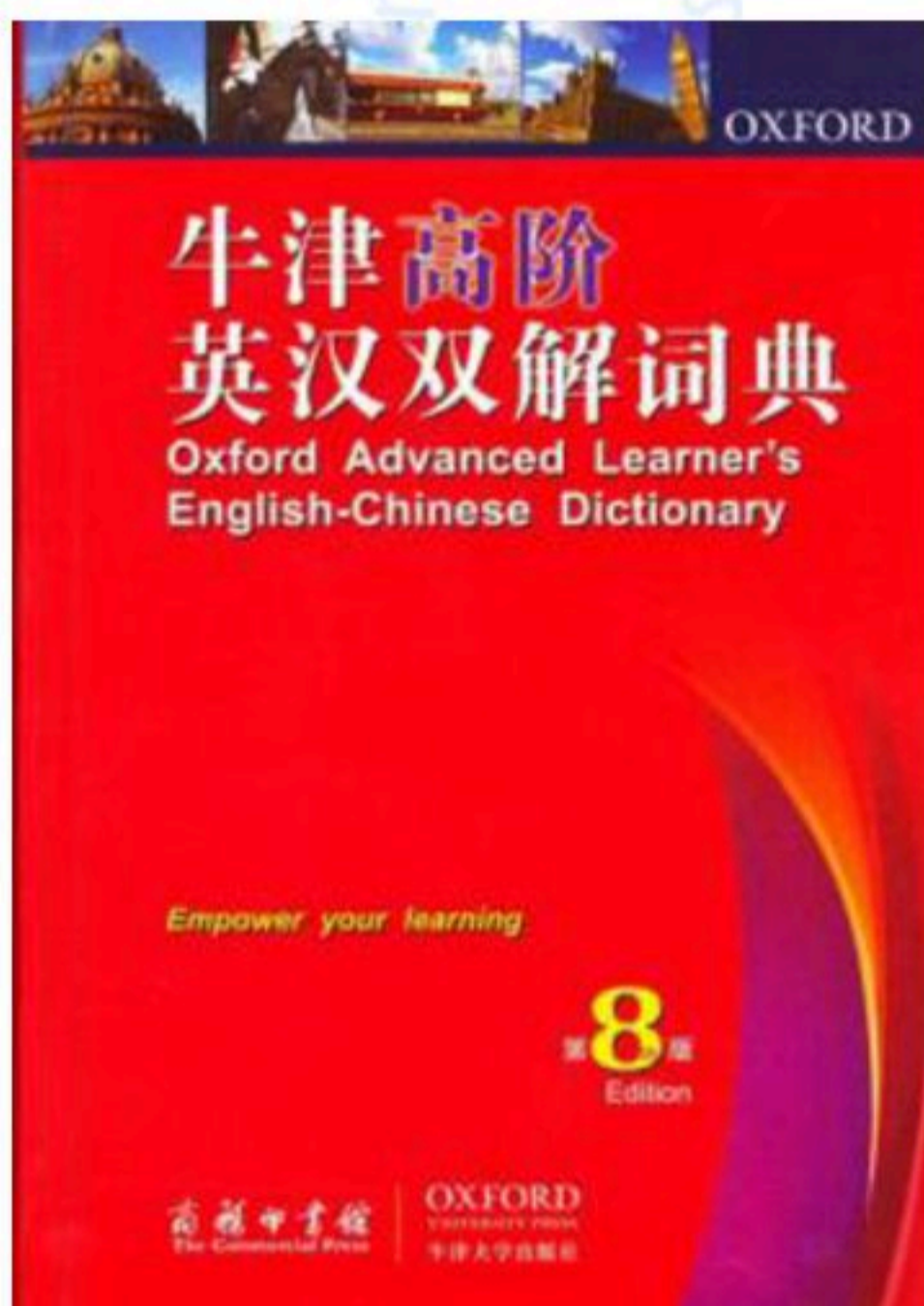




1. 探索新知

1.1

字典的创建



查字典时，每个单词都有自己对应的中文意思

python中也能够表示这种一一对应的关系，这种存储多组一一对应元素的（容器类）数据结构，在python中称为“字典”

现实中的字典

ice cream

冰淇淋

python中的字典

关键字，简称“键”，
英文“**key**”

关键字对应的内容，
称“值”，英文“**value**”

这样的一组对应关系，
叫做“**键值对**”

1.1

字典的创建

拓展：
在 Python 3.6 及更早版本中，字典是**无序的**。
在 Python 3.7 及之后的版本中，字典是**有序的**，
即插入顺序会被保留。

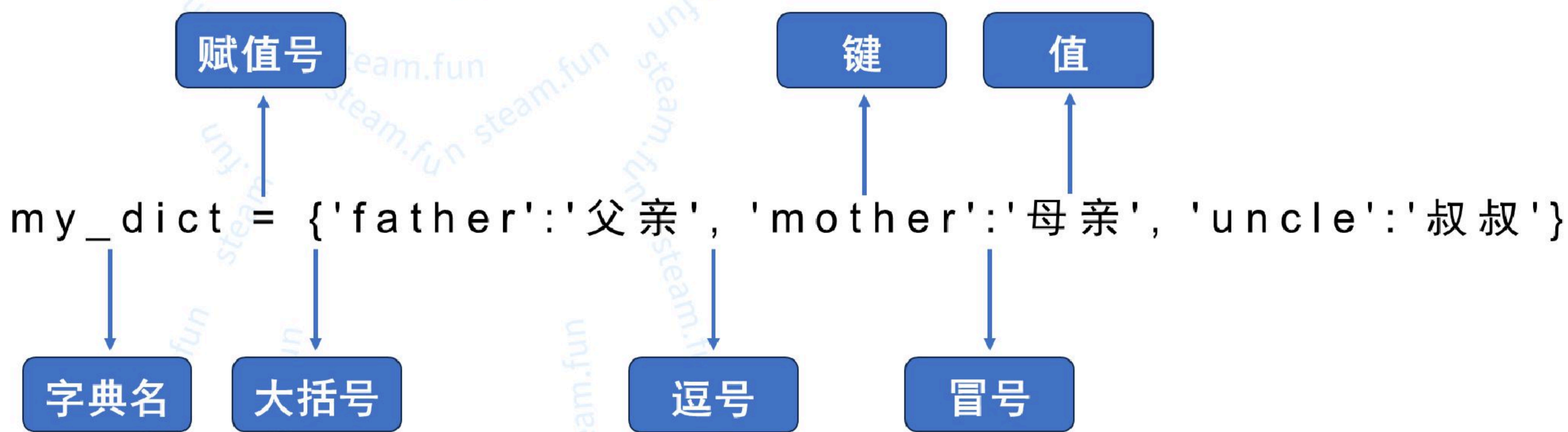
字典【dictionary】

- 字典是一种**有序**的数据结构，由键和值的**键值对**组成
- 字典中的**键必须是唯一的**，而**值可以重复**
- 字典的**键是不可变的**（通常使用字符串或数字），而**值可以是任意数据类型**

字典的创建

- 使用大括号 **{ }** 和冒号 **:** 来创建字典
- 创建空字典，`my_dict = { }`
- 创建字典一般格式：

字典变量名 = {key1: value1, key2: value2, key3: value3 ...}



注意事项

- key 和 value 之间用 (:) 隔开
- 每一组键值对间用 (,) 隔开
- 整个字典用大括号({ })括起来
- key 唯一，不可重复；value 可以重复
- key 只能是不可变数据类型，如：数字、字符串等

自己先试着创建一个简单的字典吧



1.1

字典的创建



key 键要唯一？我偏要试试，字典中如果存在两个相同的key '李白'，结果会怎样呢？

代码测试

```
my_dict = {'李白': '男生', '妲己': '女生', '李白': '诗人' }  
print( my_dict )
```

输出 _____



可以发现：
字典中的每个key只能对应一个value，
而对于同一个key，后添加的value会覆盖先前的值

除了上面的方法，还可以怎样创建字典呢？



- 使用dict()函数创建 (简单认识，以后会专门学函数)

```
my_dict = dict(name='John', age=25, city='New York')  
print(my_dict)
```

输出 _____

1.1

字典的创建

- 创建空字典

```
my_dict = dict( )
```

1.2

字典元素的访问

字典中每个元素中的“关键字-key”就相当于字典的索引。而它对应的“值-value”就是数据

- 访问字典中的值
- 修改字典中的值
- 遍历字典的键、值、键值对

接下来我们挨个儿讲讲



- 访问字典中的值

使用方括号 `[]` 和对应的 `key` 来访问字典中的值

格式为：

字典名[关键字]

具体该怎么访问呢？
代码怎么写呢？



```
my_dict = {'妲己': '法师', '后羿': '射手'}
```

```
print(my_dict['后羿']) # 输出 _____
```

```
print(my_dict['李白']) # _____
```

- 修改字典中的值

- 如果关键字存在字典中，则修改关键字对应元素的值
- 如果关键字不在字典中，则在字典中新增键值对，关键字：新的值

格式为：

字典名[关键字] = 新的值

```
my_dict = { '小明': 15733336489, '旺仔': 13511111111 }
```

```
# 修改'小明'的电话号码
```

```
my_dict[ '小明' ] = 15533339988
```

```
print(my_dict) # 输出 _____
```

```
# 添加'六子'的电话号码
```

```
my_dict[ '六子' ] = 17966664455
```

```
print(my_dict) # 输出 _____
```

- 字典元素的访问



还记得什么是遍历吗？
如何对字典进行遍历呢？

1.2

字典元素的访问

已知: `my_dict = {'name': 'John', 'age': 25, 'city': 'New York'}`

```
# 遍历字典
```

```
for i in my_dict:
```

```
    print(i)
```

```
# 输出 _____
```



难道只能遍历字典的键吗？字典的值、键值对我也想遍历，怎么操作呢？

`keys()` 方法返回字典中所有的键

`values()` 方法返回字典中所有的值

`items()` 方法返回字典中所有的键值对

通过循环遍历这些返回的结果来访问对应内容

```
# 遍历所有键
```

```
for key in my_dict.keys():
```

```
    print(key)
```

```
# 输出 _____
```

```
# 遍历所有值
```

```
for value in my_dict.values():
```

```
    print(value)
```

```
# 输出 _____
```

1.2

字典元素的访问

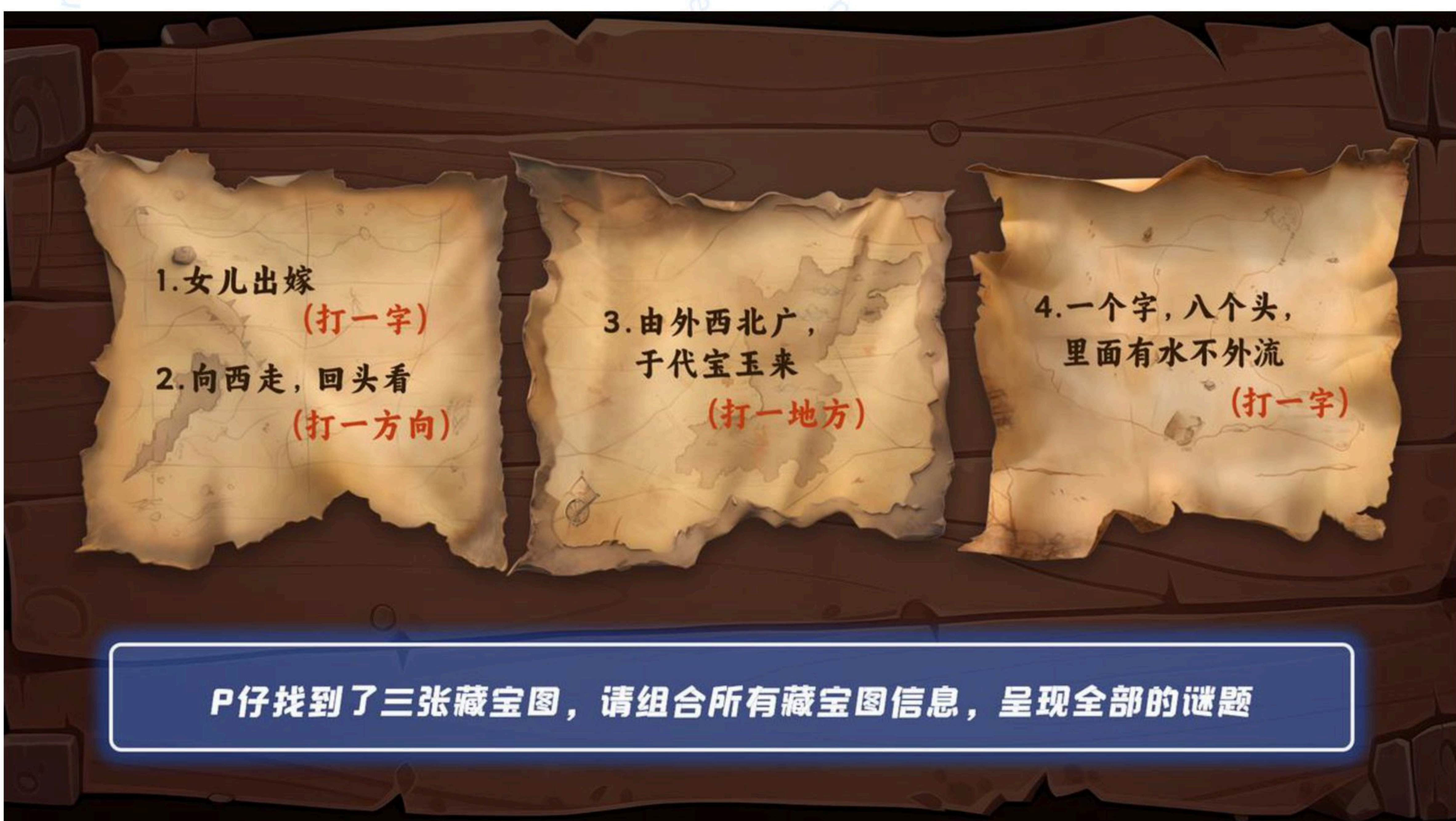
```
# 遍历所有键值对
```

```
for key, value in my_dict.items():
    print(key, value)
```

输出 _____

1.3

藏宝图现世



1. 存储三张藏宝图信息

```
# 3张藏宝图中的4个关键信息
```

```
map1 = {1: '女儿出嫁 (打一字)', 2: '向西走, 回头看 (打一方向)'}
map2 = {3: '由外西北广, 于代宝玉来 (打一地方)'}
map3 = {4: '一个字, 八个头, 里面有水不外流 (打一字)'}

```

2. 组合所有藏宝图信息

藏宝图信息此时都存储在字典中，想要组合藏宝图信息，也就是需要合并字典，我们先来学习合并字典的方法

- 字典的合并 — `update()` 方法

```
dict1 = {'妲己': '法师', '后羿': '射手'}
dict2 = {'李白': '刺客', '亚瑟': '战士'}
dict1.update(dict2) # 合并字典
print(dict1)
```

输出



试一试：

如果 `dict1 = {'妲己': '法师', '后羿': '射手'}`,
`dict2 = {'李白': '刺客', '后羿': '男性'}`,
 两个字典中存在相同键，试试合并后会如何输出？

- 组合所有藏宝图信息

串联所有的藏宝图信息，都存储在map1中

```
map1.update(map2)
```

```
map1.update(map3)
```

3. 藏宝图现世

呈现全部的谜题，即为完整的藏宝图信息

```
print(''-----开启宝藏的方法-----
下面谜底连在一起为宝藏之谜
-----')
# 遍历字典的值，输出所有谜题
for i in map1.values():
    print(i)
```

根据谜题挨个输入谜底，如果谜底全部正确，则会输出正确的藏宝地，我已经心跳加速啦！

1. 挨个询问并输入谜底

```
# 挨个询问谜底
ans1 = input('请猜测第一个谜底: ')
ans2 = input('请猜测第二个谜底: ')
ans3 = input('请猜测第三个谜底: ')
ans4 = input('请猜测第四个谜底: ')
```

2. 判断谜底正误

如果谜底全部正确, 则输出正确的藏宝地

```
if ans1=='家' and ans2=='东侧' and ans3=='庙宇' and ans4=='井' :
```

```
    print('' **** 恭喜你 ****  
成功解开了宝藏之谜 '' )
```

```
    print( '宝藏位于: 家东侧庙宇的井中' )
```

```
else:
```

```
    print( '你所猜测的地方没有宝藏' )
```

完整代码

```
# 3张藏宝图中的4个关键信息
map1 = {1:'女儿出嫁（打一字）',2:'向西走，回头看（打一方向）'}
map2 = {3:'由外西北广，于代宝玉来（打一地方）'}
map3 = {4:'一个字，八个头，里面有水不外流（打一字）'}
# 串联所有的藏宝图信息，都存储在map1中
map1.update(map2)
map1.update(map3)
# 输出所有的谜题
print(''-----开启宝藏的方法-----
下面谜底连在一起为宝藏之谜
-----'')
# 遍历字典的值，输出所有谜题
for i in map1.values():
    print(i)
# 挨个询问谜底
ans1 = input('请猜测第一个谜底：')
ans2 = input('请猜测第二个谜底：')
ans3 = input('请猜测第三个谜底：')
ans4 = input('请猜测第四个谜底：')
# 如果谜底全部正确，则输出正确的藏宝地
if ans1=='家' and ans2=='东侧' and ans3=='庙宇' and ans4=='井':
    print('**** 恭喜你 ****
成功解开了宝藏之谜')
    print('宝藏位于：家东侧庙宇的井中')
else:
    print('你所猜测的地方没有宝藏')
```



2. 强化练习

1. `d = {"王明":178, "张华":158, "于飞":189, "刘英":164}`, 则`d["于飞"]`的值是? ()

A. -2

B. 2

C. "于飞"

D. 189

2. 下列说法正确的是? ()

A. 字典只能存储字符串, 不能存储其他类型的对象

B. `dict11 = { : }`可以创建一个空字典

C. `{ 123: 456 }`是一个合法的字典

D. 字典的值必须是唯一的, 键不必是唯一的

3. 下列程序的运行结果是? ()

```
zd1 = { 'name': 'Tom', 'color': 'blue' }  
zd1[ 'name' ] = 'Mike'  
print(zd1)
```

A. `{'name': 'Mike', 'color': 'blue'}`

B. `{'name': 'Mike', 'color': 'blue', 'name': 'Tom'}`

C. `{'name': 'Tom', 'color': 'blue'}`

D. `{'name': 'Tom', 'name': 'Mike', 'color': 'blue' }`



2. 强化练习

4. 下列程序的运行结果是？（ ）

```
d1 = dict( age = 18 )  
print( d1 )
```

- A. (age = 18)
- B. [age = 18]
- C. age = 18
- D. { 'age': 18 }

5. 下列代码执行的结果是？（ ）

```
dic = { '聊天机器人': 'GhatGPT', '搜索引擎': '百度', '浏览器': 'Google' }  
for x in dic:  
    print( x, end = ' ' )
```

- A. GhatGPT 百度 Google
- B. 聊天机器人 搜索引擎 浏览器
- C. ['GhatGPT', '百度', 'Google']
- D. [('聊天机器人', 'GhatGPT'), ('搜索引擎', '百度'), ('浏览器', 'Google')]

3. 术语箱

dictionary 字典

item 项目

key 键

map 地图

value 值

update 更新

4. 课后挑战

我的小超市

已知超市的存货种类和数量如下所示：

```
shop = {'牛奶':10, '辣条':10, '方便面':8, '火腿肠':8}
```

要求如下：

- (1)有位顾客购买了4袋方便面，4根火腿肠，请更新存货数据
- (2)我的小超市最近新购入了面包5袋，请更新存货数据
- (3)有位顾客到店购物，想知道店里有哪些商品，请挨个输出店里的商品

