子涵 for 循环

```
 1    toAdd = [1]
 2    sum = 0
 3
 4    while len(toAdd) <=99:
 5        toAdd.append(toAdd[-1]+1)
 6
 7    for element in toAdd:
 8        sum = sum + element
 9
10    print(sum)
```

子涵 for 循环 偶数

```
1  sum = 0
2
3  for i in range(2,101,2):
4      sum = sum + i
5
6  print(sum)
7
```

显皓：range里不能有变量，step所以也必须是常量

```
>>> for i in range(0, 10, i):
...        print(i)
...
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'i' is not defined
>>>
```

```python
# 求: 1 到 100 的和
# for 循环 和 while 循环各实现一个

n = 100

# for loop
sum = 0
for i in range(1, n+1):
    # sum = sum + i
    sum += i
print('for loop result:', sum)

# while loop
sum = 0
i = 1
while i <= n:
    sum += i # sum = sum + i
    i += 1
print('\nwhile loop result:', sum)
# '\' escape next letter 跳过下一个字符, 即下一个字符不作为字符串
# 而有新意义 '\n' new line 换行


# 求: 1 到 100 偶数的和
# for 循环 和 while 循环各实现一个

print('\n偶数一共有: ' + str(len(range(2, n+1, 2))) + '个')
# range(): 指定一个值的范围, 给出这个范围内的值的列表, 这个列表可循环
# len(): 给出列表的长度

# for loop
sum = 0
for i in range(2, n+1, 2):
    sum += i
print('\nfor loop 偶数 result:', sum)

# while loop
sum, i = 0, 1
while i <= n:
    if i % 2 == 0:
        sum += i
    i += 1
print('\nwhile loop 偶数 result:', sum)
```

```
(base) pz loops_contditionals_examples $ python loop_conditional_quiz.py
for loop result: 5050

while loop result: 5050

偶数一共有：50个

for loop 偶数  result: 2550

while loop 偶数  result: 2550
(base) pz loops_contditionals_examples $ 
```

Python's range() Function Examples

Simple Usage

```
1  >>> # One parameter
2  >>> for i in range(5):          range(5): [0, 1, 2, 3, 4]
3  ...     print(i)
4  ...
5  0
6  1
7  2
8  3
9  4
10 >>> # Two parameters
11 >>> for i in range(3, 6):
12 ...     print(i)
13 ...
14 3
15 4
16 5          element      iterable: - list, - range(…)
17 >>> # Three parameters
18 >>> for i in range(4, 10, 2):          for i in [0, 1, 2]:
19 ...     print(i)
20 ...
21 4                  range(start, end, step)
22 6
23 8
24 >>> # Going backwards
25 >>> for i in range(0, -10, -2):
26 ...     print(i)
27 ...
28 0
29 -2
30 -4
31 -6
32 -8
```

Python 3.x | Python 2.x

```
1  >>> range(1)
2  range(0, 1)
3  >>> type(range(1))
4  <class 'range'>
```

Python 3.x

```
>>> list_of_ints = list(range(3))
>>> list_of_ints
[0, 1, 2]
```

default start and step: 0