Try else


```
1 ————————
2 ————————
3 ————————
try :
    4 ————————
    5 ————————
    6 ————————
except :
    7 ————————
else :
    8 ————————
9 ————————
10 ————————
```

Try-except-else

As we already know we use for else, while else.

For…….
……..
Else:
……
In for loop this else block is executed. If it is not breaking in between.

If for loop is stopping with break statement then else block will not be executed.

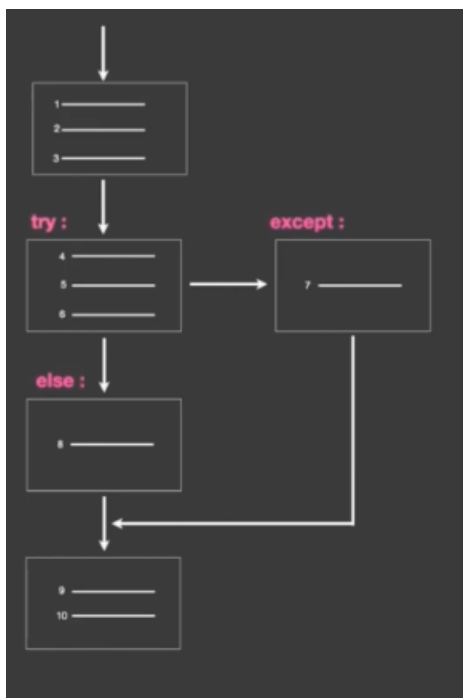While as long as the condition written… in while loop is false
Else: go to else condition

In both the conditions if for loop gets executed. Successfully the else will execute

 In both the conditions if while loop gets executed. Successfully the else will execute

Same way if try block is executing without any exception raised then else block is executed

It means if there is any exception in try block then except block will execute and else will not be execute.

Else block confirms that try block executed successfully

```python
print('before try')

try:
    a = int(input('Enter numerator'))
    b = int(input('Enter Denominator'))
    c = a // b
    print('Try Block executed successfully')
except ZeroDivisionError as err:
    print(err)
else:
    print('Division is', c)

print('outside try-except-else')
```