HD EDU

INFO1110
Introduction to Programming

同步辅导课5

Tutor：Hilbert

HD·EDUCATION
全球累计服务用户超十万

HD EDU



HD EDU

企业介绍

海道教育（HD EDU）成立于2018年，是一家助力国际化人才成长的教育科技企业，同时也是一个力求突破语言和文化障碍、推动国际教育与文化属性相融合的新锐品牌。

海道教育已实现全球规模化发展，旗下产品已走入英国、美国、澳大利亚、新西兰等主流留学的国家40多所院校，为超30万的中国留学生提供高品质的教学服务，成为留学后市场在线教育的领军企业。

2022年，HD EDU发布首个学术诚信合规准则，HD EDU愿始终与所有用户一起，坚守学术尊严，真正意义上打造既保护用户权益又懂用户需求的教学产品，推动海外华人留学生成长与发展。

产品介绍

成立五年来，HD EDU 作为专注于留学后市场服务的后起之秀，秉持着“提升中国留学生核心竞争力”的产品理念，深度洞察留学生教育场景叠加、需求变化，研发多阶段教育产品，为用户提供多元化、高品质的一站式留学教育服务，打造出符合留学生全生命周期的三大教育产品线：HD Tutoring、HD Career、HD Offer。

HD Tutoring

学习有海道，提分更有效

节点辅导　全学期学业辅导

HD Career

让年轻人找到喜欢的工作

技能课程　资格证书　实训项目　求职方案

HD Offer

专注藤校G5级别院校申请

选校定位　文书创作　申请服务　面试辅导


HD EDU
以用户的成长为第一

HD EDU

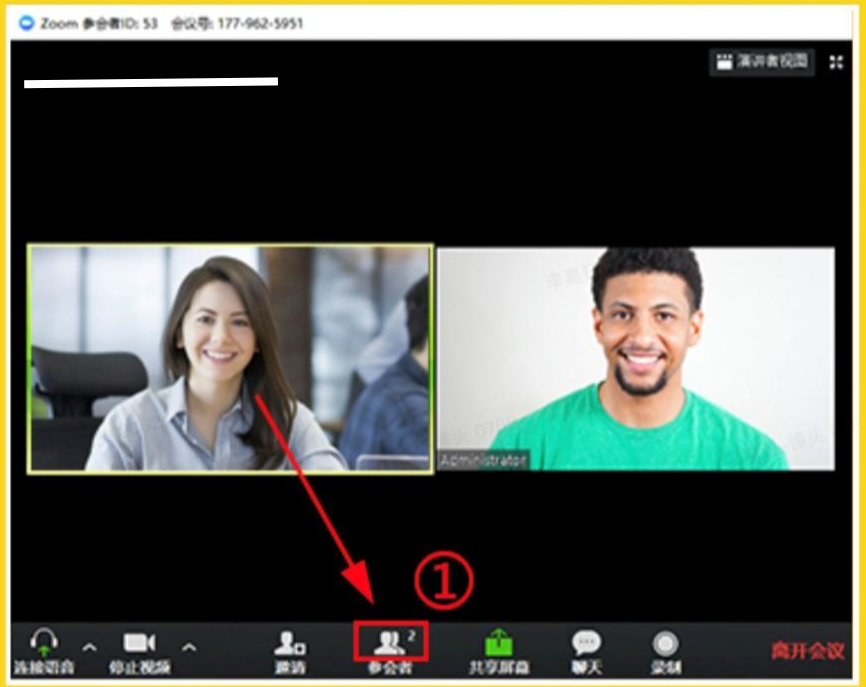


TUTOR Hilbert

Self- Introduction

- USYD Bachelor of Advanced Computing Honours First Class毕业 (Major in Software Development)
- 東京工業大学Master of Artificial Intelligence在读
- 曾任USYD INFO1110的Lab Tutor
- INFO1110总分91.0 HD
- 2018年加入HD Edu，主要教授预科及本科的IT入门课程
- HD Edu学科带头人兼面试官，曾经撰写过HD的INFO课程规范指南

同学们遇到问题怎么办？







1. 点【参会者】
2. 点【举手】即可与老师实时互动
3. 问题被解答了还可以【手放下】
4. 点【聊天】/【Chat】就可以与老师实时文字互动哦

直播平台互动方法

直播平台 ： 举手+聊天室提问



1. 点【参会者】再点【举手】，即可与老师实时互动！在此输入你想问的问题
2. 问题被解答了还可以【手放下】

本章知识点

1. File I/O
2. File Names and Extensions *
3. os Module *
4. Testing
5. CSV
6. Shell Redirection *



HD EDU



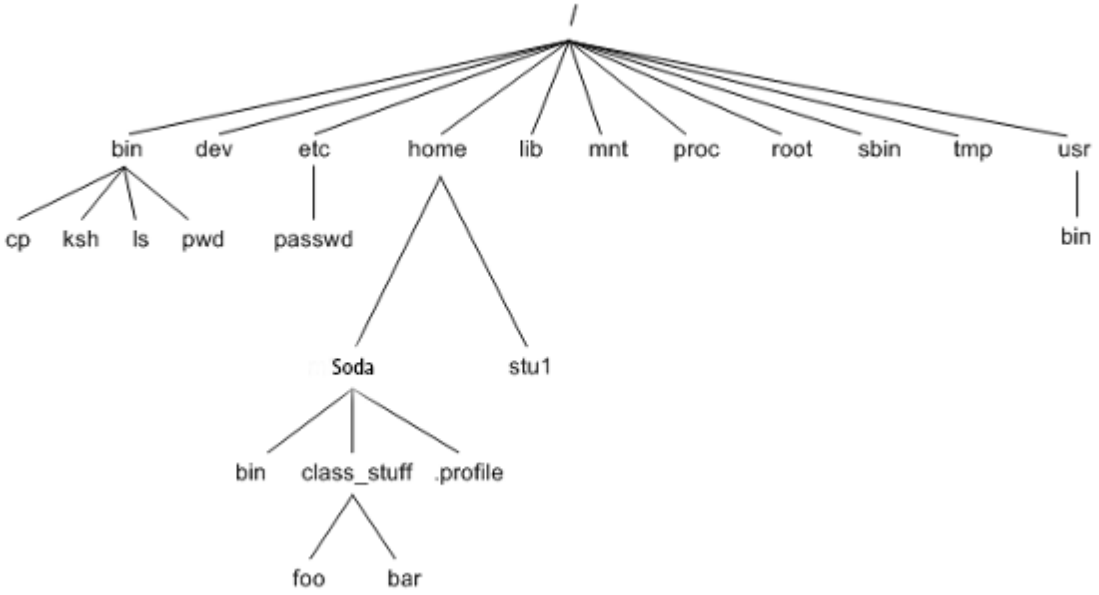
知识点讲解

考点1. File I/O

HD EDU

- So far, 我们已经学了一下几个方式给我们的程序input，以及让程序输出output:
- Inputs:
  - `input()` function: Read from stdin
  - Command-line arguments
    - `import sys; sys.argv`
- Outputs:
  - `print()` function: Print to stdout
- 而学完这个章节，我们会额外得到两个新的方式分别是从文件读取inputs和output到文件

考点1. File I/O - Path



Absolute Path vs **Relative** Path

- Absolute: You can access the same file from any location. E.g. /bin/sh
- Relative: You can access the file based on current location e.g., soda/Downloads

❖ 当你用python file.py运行你的程序的时候，**你的working directory就在这个file.py所在的位置**

考点1. File I/O – open( ) function

HD EDU

```
f = open(path: str, mode='r': str)
```

'r', 'w', 'a' => Read, Write, Append

```
# Some IO operations

…

…

…
```

f.close( ) <- 这是一个method

```
>>> f = open('1.txt')
>>> type(f)
<class '_io.TextIOWrapper'>
>>> f.read()
'Hello World!\nHello USYD\n'
>>> f.close()
>>> f = open('1.txt', 'r')
>>> f.read()
'Hello World!\nHello USYD\n'
>>> f.close()
```

❖注意有开就要有关!

考点1. File I/O – Reading

HD EDU

❖ 当你读文件的时候，最重要的一个点在于有这个文件可以读，否则:

```
>>> f = open('2.txt', 'r')
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
FileNotFoundError: [Errno 2] No such file or directory: '2.txt'
```

❖ 所以，比较常见的读文件的结构一般会和try用在一起:

```
try:
  f = open('2.txt', 'r')
except FileNotFoundError:
  # 做一些补救措施，如:
  print("The file cannot be found!")
```

考点1. File I/O – Read Methods

HD EDU

Reading methods:

`f.read()`

`f.readline()`

`f.readlines()`

`f.read().splitlines()`

```
>>> f = open("lorem.txt")
>>> f.read()
'Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusm
od tempor incididunt ut labore et dolore magna aliqua.\nUt enim ad min
im veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip e
x ea commodo consequat.\nDuis aute irure dolor in reprehenderit in vol
uptate velit esse cillum dolore eu fugiat nulla pariatur.\nExcepteur s
int occaecat cupidatat non proident, sunt in culpa qui officia deserun
t mollit anim id est laborum.\n'
>>> f.close()
```

```
>>> f = open("lorem.txt")
>>> f.readlines()
['Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eius
mod tempor incididunt ut labore et dolore magna aliqua.\n', 'Ut enim a
d minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliq
uip ex ea commodo consequat.\n', 'Duis aute irure dolor in reprehender
it in voluptate velit esse cillum dolore eu fugiat nulla pariatur.\n',
'Excepteur sint occaecat cupidatat non proident, sunt in culpa qui off
icia deserunt mollit anim id est laborum.\n']
>>> f.close()
```

```
>>> f = open("lorem.txt")
>>> f.readline()
'Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusm
od tempor incididunt ut labore et dolore magna aliqua.\n'
>>> f.readline()
'Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris ni
si ut aliquip ex ea commodo consequat.\n'
>>> f.close()
```

考点1. File I/O – Read Methods

HD EDU

Reading methods:

`f.read()`：读整个文件，return一个string （Pointer移到文件尾)

`f.readline()` ：读一行，return一个string（Pointer移到下一行开头)

`f.readlines()` ：读整个文件，每行是一个string，return一个list （Pointer移到文件尾)

`f.read().splitlines()`

- 注意New line character!
- 注意有光标的概念

```
>>> f = open('lorem.txt', 'r')
>>> f.read()
'Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusm
od tempor incididunt ut labore et dolore magna aliqua.\nUt enim ad min
im veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip e
x ea commodo consequat.\nDuis aute irure dolor in reprehenderit in vol
uptate velit esse cillum dolore eu fugiat nulla pariatur.\nExcepteur s
int occaecat cupidatat non proident, sunt in culpa qui officia deserun
t mollit anim id est laborum.\n'
>>> f.read()
''
>>> f.close()
```

考点1. File I/O – Pointer

HD EDU

```
>>> f = open('lorem.txt', 'r')
>>> f.readline()
'Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusm
od tempor incididunt ut labore et dolore magna aliqua.\n'
>>> f.readline()
'Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris ni
si ut aliquip ex ea commodo consequat.\n'
>>> f.readline()
'Duis aute irure dolor in reprehenderit in voluptate velit esse cillum
dolore eu fugiat nulla pariatur.\n'
>>> f.readline()
'Excepteur sint occaecat cupidatat non proident, sunt in culpa qui off
icia deserunt mollit anim id est laborum.\n'
>>> f.readline()
''
>>> f.close()
```

```
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmo
d tempor incididunt ut labore et dolore magna aliqua.
Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nis
i ut aliquip ex ea commodo consequat.
Duis aute irure dolor in reprehenderit in voluptate velit esse cillum
dolore eu fugiat nulla pariatur.
Excepteur sint occaecat cupidatat non proident, sunt in culpa qui offi
cia deserunt mollit anim id est laborum.
```

```
>>> f = open('lorem.txt', 'r')
>>> f.read()
'Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusm
od tempor incididunt ut labore et dolore magna aliqua.\nUt enim ad min
im veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip e
x ea commodo consequat.\nDuis aute irure dolor in reprehenderit in vol
uptate velit esse cillum dolore eu fugiat nulla pariatur.\nExcepteur s
int occaecat cupidatat non proident, sunt in culpa qui officia deserun
t mollit anim id est laborum.\n'
>>> f.read()
''
>>> f.seek(0)
0
>>> f.read()
'Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusm
od tempor incididunt ut labore et dolore magna aliqua.\nUt enim ad min
im veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip e
x ea commodo consequat.\nDuis aute irure dolor in reprehenderit in vol
uptate velit esse cillum dolore eu fugiat nulla pariatur.\nExcepteur s
int occaecat cupidatat non proident, sunt in culpa qui officia deserun
t mollit anim id est laborum.\n'
>>> f.close()
```

- `f.seek(0)`：把Pointer移到文件开头

考点1. File I/O – str.strip( )

HD EDU

- str.strip( ): 把str开头和结尾处所带的空格字符（包含空格及特殊字符如\n, \t等） “脱”去
- str.lstrip( ): Strip left
- str.rtrip( ): Strip right

```
>>> f = open("lorem.txt")
>>> f.readline()
'Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusm
od tempor incididunt ut labore et dolore magna aliqua.\n'
>>> f.readline().strip()
'Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris ni
si ut aliquip ex ea commodo consequat.'
>>> f.close()
```

- 注意strip只会看**两边**

```
>>> s = "\t \nABCD \t\nEFG\n \t"
>>> s.strip()
'ABCD \t\nEFG'
>>> s.lstrip()
'ABCD \t\nEFG\n \t'
>>> s.rstrip()
'\t \nABCD \t\nEFG'
```

- str.splitlines( ) = str.split(“\n”)

考点1. File I/O – Stream

HD EDU

- Stream I/O (Input/Output) refers to the concept of handling data as a continuous stream of bytes or characters, rather than reading or writing data all at once.
- It allows for **efficient** data processing, especially when dealing with large amounts of data or data that is being transmitted over a network.
- 虽然但是，我建议大家在INFO1110涉及到文件Reading都**优先采取把文件完整内容读完存进一个variable里，然后接下来所有的操作都通过这个variable来做。**
- 因为INFO1110所有打交道的文件都不大，全部读进内存里可以省去很多麻烦事。

考点1. File I/O – Writing

HD EDU

- 当你写文件的时候，就不存在FileNotFoundError的情况
- 如果文件不存在，Python会自动建一个文件并写入write的内容
- 如果文件存在：
  - “w”: Overwrite覆盖
  - “a”: Append续写

Write methods记得一定要自己加\n否则不会换行

```
f.write(str) -> return int(Number of characters written)

f.writelines(list) -> return None
```

- Write写的内容都会暂存在buffer里，只用等到f.close()之后才会flush到对应的文件。

考点1. File I/O – with statement

```
<var> = open("file.txt", 'r')
...
<var>.close()
```



```
with open("file.txt", 'r') as <var>:
  ...
  # <var>的lifecycle仅存在于这个indentation block
```

- 用with就不需要专门去call close( ) method。Python会自动帮你关闭。
- 但是with是只有Python才有的Syntax，其它编程语言没有
- 我INFO1110最喜欢最喜欢用的读文件操作：

```
try:
  with open('file.txt', 'r') as f:
    txt = f.read().splitlines()
except FileNotFoundError:
  # Do something
print(txt)
```

之后只需要读txt这个variable就可以啦！

考点2. File names and their extensions *

HD EDU

- The file extension is a **convention** that provides a **hint** about the file type and its intended use, allowing operating systems and applications to associate files with appropriate programs and utilities. **However, the actual content and format of a file are not solely determined by its extension.**
- 所以，把一张图片的后缀名从.jpg改成.pdf并不就意味着你把一张图片转换成了PDF文档。只是改了文件名，不会影响文件本身存的data和format。

```
$ echo "print('Hello world')" > hello.jpg
$ python3 hello.jpg
Hello world
```

考点3. os module *

HD EDU

- `os.getcwd()` - Return a string representing the current working directory.
- `os.listdir(path: str)` - Return a list containing the names of the entries in the directory given by path.

- `os.path.exists(path: str)` - Check if the path exists
- `os.path.isdir(path: str)` - Check if the path is a directory
- `os.path.idfile(path: str)` - Check if the path is a regular file
- `os.path.abspath(path: str)` - Return the absolute path of the given path

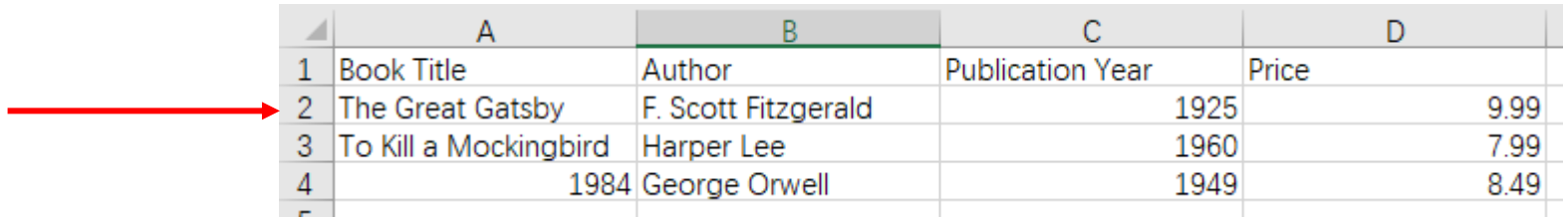
Official Doc

考点4. CSV

HD EDU

- CSV stands for **Comma-Separated Values**. It is a simple file format used to store tabular data, such as spreadsheets or databases, in a plain text format.

```
Book Title,Author,Publication Year,Price
"The Great Gatsby",F. Scott Fitzgerald,1925,9.99
"To Kill a Mockingbird",Harper Lee,1960,7.99
"1984",George Orwell,1949,8.49
```



	A	B	C	D
1	Book Title	Author	Publication Year	Price
2	The Great Gatsby	F. Scott Fitzgerald	1925	9.99
3	To Kill a Mockingbird	Harper Lee	1960	7.99
4	1984	George Orwell	1949	8.49

- Csv靠comma来分列，靠\n来分行
- 那如果数据里面就有逗号怎么办?

	A	B	C	D
1	Book Title	Author	Publication Year	Price
2	The Great Gatsby	F. Scott Fitzgerald	1925	9.99
3	To Kill a Mockingbird	Harper Lee	1960,	7.99
4	1984	George Orwell	1949	8.49

考点5. Testing – Test-driven Development

HD EDU

Testing can prove the presence of bugs, but not their absence. ——Edsger Wybe Dijkstra

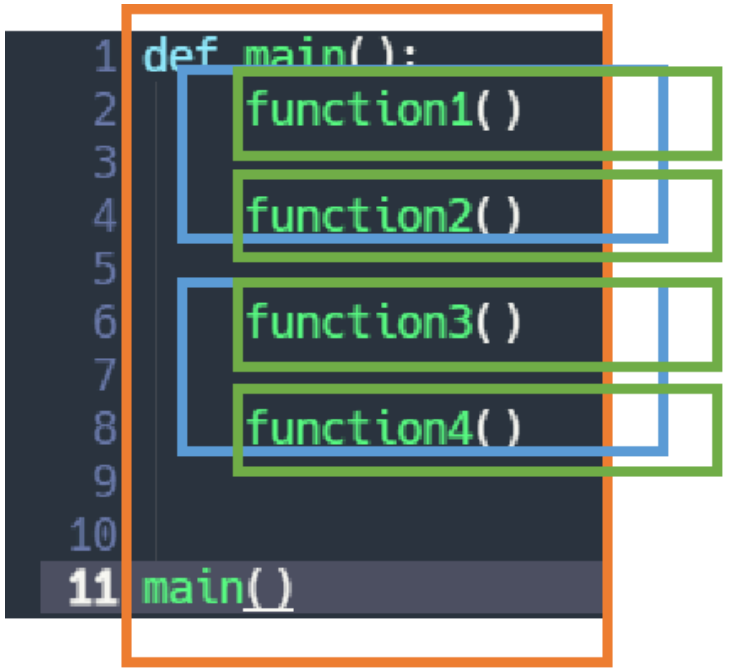
Test-Driven Development (TDD) is a software development process that relies on the repetition of a very short development cycle: first, the developer writes an (initially failing) automated test case that defines a desired improvement or new function, then produces the minimum amount of code to pass that test, and finally refactors the new code to acceptable standards.

- ❖ Write tests first
- ❖ Regression Testing

考点5. Testing – The Testing Pyramid

HD EDU

- **End-to-end test**: test the functionality and performance of an application under product-like circumstances and data to replicate live settings.
- **Integration test**: in which the different units, modules or components of a software application are tested as a combined entity.
- **Unit test**: in which the smallest testable parts of an application, called units, are individually and independently scrutinized for proper operation



考点5. Testing – Different Types of Testcases

HD EDU

- Positive Case
  - Testing the software with valid inputs and expected conditions. The goal is to verify that the software behaves correctly when provided with correct and valid data, and that it produces the expected output or behavior.
- Negative Case
  - Testing the software with invalid, unexpected, or extreme inputs and conditions. The purpose is to ensure that the software can handle errors, exceptions, and edge cases gracefully, without crashing or producing unexpected behavior.
- Edge Case occurs at an extreme (maximum or minimum) operating parameter.
- Corner Case is when multiple parameters are simultaneously at extreme levels.

考点5. Testing – Edge Case and Corner Case

HD EDU

- An edge case is an issue that occurs **at an extreme** (maximum or minimum) operating parameter.
  一个维度的边界

- A corner case is when **multiple parameters** are **simultaneously** at extreme levels.
  多个维度同时到达边界





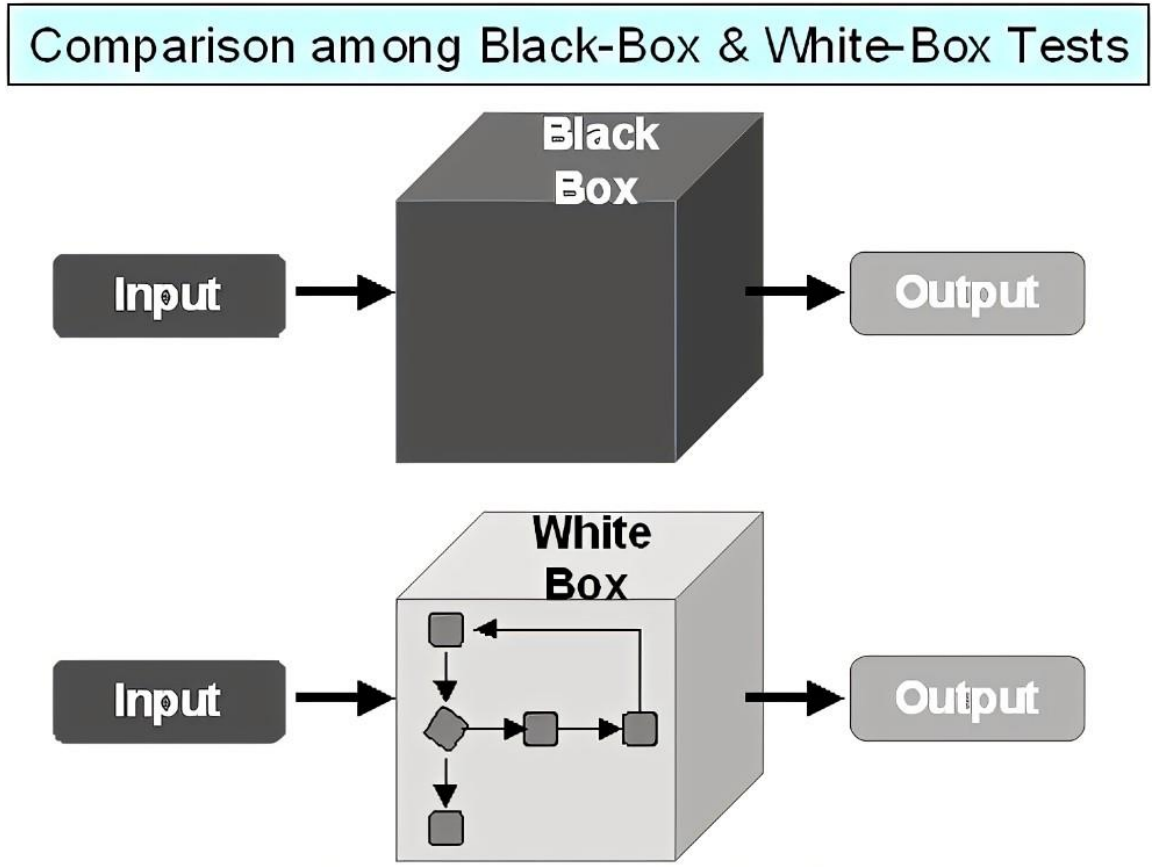


Reference: https://www.educative.io/answers/corner-case-vs-edge-case

考点5. Testing – Black-box/White-box Testing

HD EDU

- Black-box Testing
  - Black-box testing, also known as functional testing or opaque testing, is a testing approach that involves testing the software from an external or end-user perspective, **without having any knowledge of the internal structure, design, or implementation details of the software**.
- White-box Testing
  - White-box testing, also known as structural testing, glass-box testing, or clear-box testing, is a testing approach that **involves examining the internal structure, design, and implementation details of the software**.



考点5. Testing - Assert

HD EDU

```
assert <boolean expression> [, msg(str)]
AssertionError
```

```
def func_to_test(a, b, c):
  ...
  ...
  return ...
```

```
from <file> import func_to_test

def testcase1():
  a = xx
  b = xx
  c = xx
  expected = xx
  actual = func_to_test(a, b, c)
  assert expected == actual, "Test 1 Faile
d!"
```

考点5. Testing - Testcase设计攻略

HD EDU

```python
def func(a: int, b: str, c: list[]) -> bool:
  pass
```

- Unit Test里面优先看function signature
- Int/float
  - 正/副/零/特殊值
- Str
  - “” Empty string
  - “ “
  - “ldsjfklsj”
- Reference Datatype
  - None
- Boolean
  - True/False

- 保证测试范围覆盖广
- 变量排列组合

考点6. Shell Redirection *

HD EDU

- Input Redirection <
- Output Redirection >
- Pipe

```
$ cat 2.txt
Hello
$ cat temp.py
a = input()

for i in range(5):
    print(a)
$ python temp.py < 2.txt
Hello
Hello
Hello
Hello
Hello
$ python temp.py > 3.txt
Hello
$ cat 3.txt
Hello
Hello
Hello
Hello
Hello
$ python temp.py | wc
Hello
      5       5      30
```



经典题型放送

Bash Diff Test Example

HD EDU

+ test.sh
+ main.py
+ inputs
+ - - 1.in
+ - - 2.in
+ outputs
+ - - 1.out
+ - - 2.out

```
$ cat test.sh
#! /usr/bin/env sh

count=0
for test in inputs/*.in; do
        name=$(basename $test .in)
        expected=outputs/$name.out
        python3 main.py < $test | diff - $expected || echo "Tes
t $name: failed!\n"
        count=$((count+1))
done

echo "Finished running $count tests!"
```

(自编题)

- 这种测试**必须保证**main.py里面有input()接受输入以及print()给出输出

Black/White Box Testing (自编题)

HD EDU

`count(li, s)` returns the number of occurrences of s in li.

`s:` a character

`li:` a list of strings

Testcase ID	Inputs	Expected Output
1		
2		
3		
4		

Black/White Box Testing

(自编题)

HD EDU

```
def count(li, s):
  if not isinstance(li, list):
    raise TypeError
  if not isinstance(s, str):
    raise TypeError
  if len(s) != 1:
    raise SyntaxError

  i = 0
  while i < len(li):
    current = li[i]
    if isinstance(current, str):
      if len(current) != 1:
        raise SyntaxError
      else:
        if current == s:
          count += 1
        i += 1
    else:
      raise TyperError
  return count
```

Testcase ID	Inputs	Expected Output
1		
2		
3		
4		

Class and File I/O （自编题）

HD EDU

Write a class named `Student(name, age, gender, unikey, SID, phone)`.

- `name` - (str) the name of the student
- `age` - (int) the age
- `gender` - (str)
- `unikey` - str
- `SID` - (int) **Must be** 9-digit, throw `ValueError` otherwise.
- `phone` - (str)

Implement a method `export()` which will write all the information to a file. (Filename is the student's name + `.txt`)

```
# ValueError
a = Student('John', 20, 'male', 'abcd1234', 48008216, "+61 123346674")

a = Student('John', 20, 'male', 'abcd1234', 480082164, "+61 123346674")
a.export()
```

Class and File I/O （自编题）

HD EDU

John.txt:

```
Name: John
Age: 20
Gender: male
Unikey: abcd1234
SID: 480082164
Phone: +61 123346674
```

Code – Sort

(自编题)

Write a function `sort(li)` that receives a list of integers and return a sorted list in ascending order.

- Assume the list does not contain duplicates.

```
sort([4, 5, 2, 8, 1, 0, 3, 7] -> [0, 1, 2, 3, 4, 5, 7, 8]
```

Intuitive Sort with Min

HD EDU

```
Function my_sort(collection):
        new_collection = []
        i = 0
        while i < len(collection):
                temp = find 第i小的element in collection
                new_collection.append(temp)
        return new_collection
```

Sort – in-place

HD EDU

```
Function bubble_sort(collection):

        i = 0

        while i < length of the collection then

                j = 0

                while j < length of the collection - i - 1 then

                        if collection[j] > collection[j+1] then

                                swap values between collection[j] and collection [j+1]

                        j += 1

                i += 1

        return
```

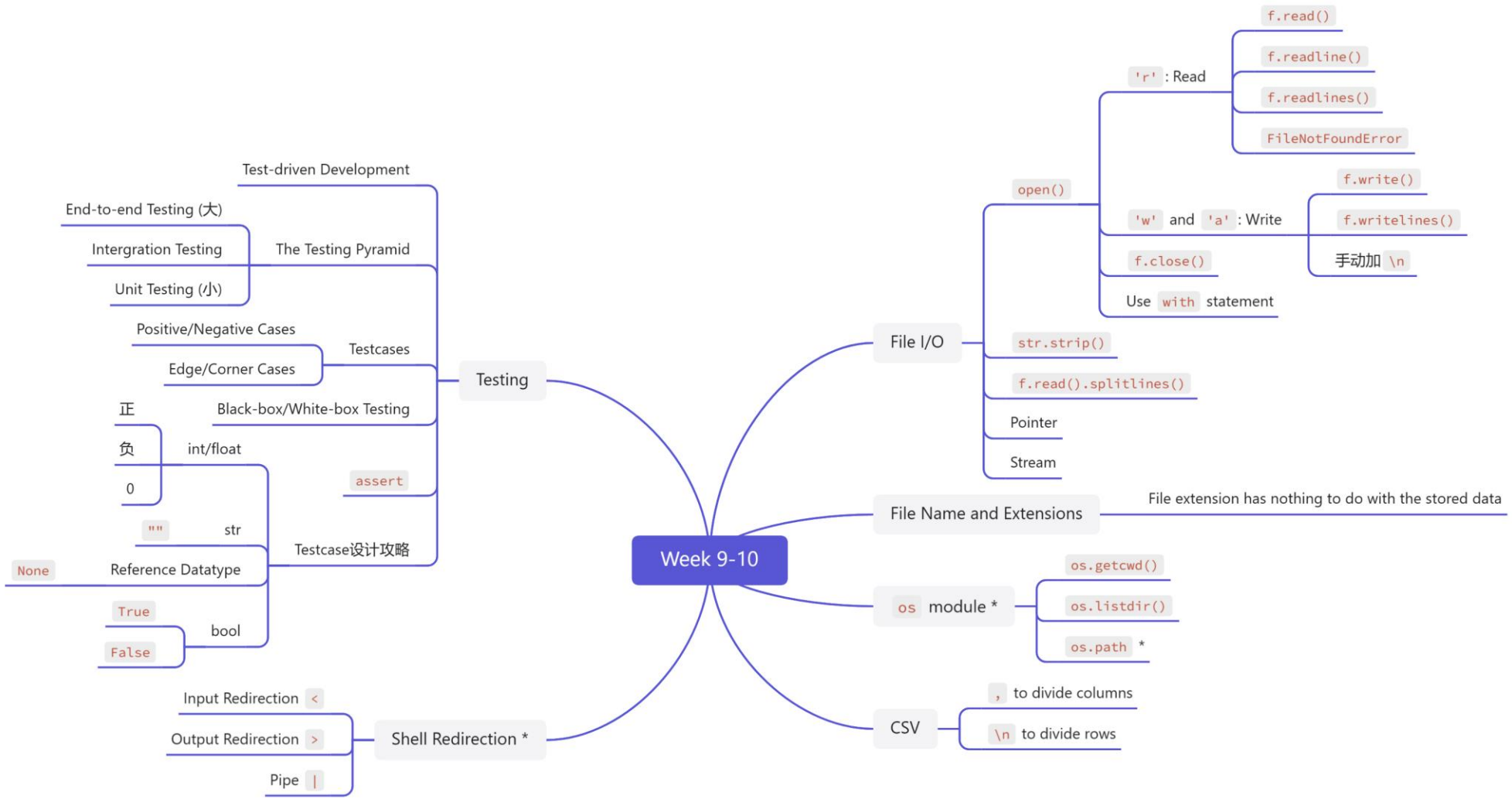
Descending order?





重难点总结

重难点总结


Week 9-10
Testing
Test-driven Development
The Testing Pyramid
End-to-end Testing (大)
Intergration Testing
Unit Testing (小)
Testcases
Positive/Negative Cases
Edge/Corner Cases
Black-box/White-box Testing
assert
Testcase设计攻略
int/float
正
负
0
str
""
Reference Datatype
None
bool
True
False
Shell Redirection *
Input Redirection <
Output Redirection >
Pipe |
File I/O
open()
'r' : Read
f.read()
f.readline()
f.readlines()
FileNotFoundError
'w' and 'a' : Write
f.write()
f.writelines()
手动加 \n
f.close()
Use with statement
str.strip()
f.read().splitlines()
Pointer
Stream
File Name and Extensions
File extension has nothing to do with the stored data
os module *
os.getcwd()
os.listdir()
os.path *
CSV
, to divide columns
\n to divide rows



下节课预告

下节课预告

HD EDU

- Recursion
  - What will happen if a function calls itself?
- Iterator
  - What is actually going on in for-each loop?



您每一份诚挚的问卷都会拯救一根濒临掉落的头发。

HD EDU



帮我开个张吧，求求你了！



课程结束后，如果您对课程或者服务的任何建议和意见

请给予我们提高和改进的机会，感谢您对 HD·EDUCATION 课程和服务的信任！

· 填写问卷操作流程 ·



第一步

关注【海道教育】服务号后
点击【购买通知】或【上课提醒】



第二步

点击【前往学习】



第三步

点击【去评价】就可以为课程进行评价



Q & A