**Python 面試題目**

1. 請舉例三個程式開發上使程式碼好維護的原則

|  |
| --- |
| 默想：我現在這樣寫，不久後我或別人看得懂嗎？   1. 加上註解，提醒自己或告知其他使用者這段程式碼的用意，註解最好能簡明扼要。 2. 命名語意要能讓人快速理解，可參考蛇行命名法與駝峰式命名法。 3. 製作說明文件，最好能圖文並茂（附上表格為佳），並說明引用的套件有哪些、及操作流程。 4. 寫法盡量統一，雖然有多種解法但盡量只使用一種。比方說讀Excel檔案，可以使用xlrd或pandas，但盡量擇一使用;或者import datetime / from datetime import datetime 擇一使用。 5. 不要把程式碼寫死，且預測可能發生的變化。縱使現在程式可以執行，但若資料型態不同，是否會有Bug。 |

1. 請舉例並說明一個使用過的裝飾器其功能為何

|  |
| --- |
| Flask中為路由（Route），說明如下。   1. 透過函式的裝飾器設定路由 2. 決定網址路徑和處理函式的對應關係，前端輸入不同路徑時，後端程式要決定對應的處理函式。 3. 範例如下：（完整程式碼請參考GitHub：<https://github.com/AcerPing/Ace_Pycharm_Project_2020-08-04Renewed/blob/master/Flask_%20MessageBoard/index.py>）   @app.route(**'/'**) def hello():  page = request.args.get(**'page'**, default=1, type=int)  print(**"請求方法"**, request.method)  print(**"頁數："**,page)  list = load\_data\_page(page)return render\_template(**"index.html"**, list=list) |

1. 請定義一個function名為append\_to\_list，功能說明如下：

|  |
| --- |
| 傳入參數為n, list, 當list沒提供的時候自動用空陣列作為default  此function功能為 將n附加到list最後傳出  append\_to\_list(n=1)  >>> [1]  append\_to\_list(n=1, lst=[1,2])  >>> [1,2,1] |
| def append\_to\_list(n,lst=[]):  lst.append(n)  return lst  lst = append\_to\_list(n=1)  print(lst)  lst = append\_to\_list(n=1, lst=[1,2])  print(lst) |

1. 已知網頁API回應內容如下

|  |
| --- |
| {  "total": 5,  "data": [  {"id": "001", "name": "productA", "price": 25, "category": "AAA", "stock": 7},  {"id": "002", "name": "productB", "price": 30, "category": "AAA", "stock": 17},  {"id": "003", "name": "productC", "price": 60, "category": "BBB", "stock": 2},  {"id": "004", "name": "productD", "price": 10, "category": "CCC", "stock": 26},  {"id": "005", "name": "productE", "price": 15, "category": "BBB", "stock": 16},  ]  }  *# 請按照category欄位進行整理，算出每個category的stock加總：*  {  "AAA": {"products": ["ProductA", "ProductB"], "stock": 24},  "BBB": {"products": ["ProductC", "ProductE"], "stock": 18},  "CCC": {"products": ["ProductD"], "stock": 26}  } |
| # 題目  API\_Response = {"total": 5,  "data": [  {"id": "001", "name": "productA", "price": 25, "category": "AAA", "stock": 7},  {"id": "002", "name": "productB", "price": 30, "category": "AAA", "stock": 17},  {"id": "003", "name": "productC", "price": 60, "category": "BBB", "stock": 2},  {"id": "004", "name": "productD", "price": 10, "category": "CCC", "stock": 26},  {"id": "005", "name": "productE", "price": 15, "category": "BBB", "stock": 16},]}  # Solution  Result = {}  for each\_id in API\_Response["data"]:  # print(each\_id["category"])  if each\_id["category"] not in Result:  Sub\_Result = {}  Sub\_Result.setdefault("products", [each\_id["name"]])  Sub\_Result.setdefault("stock", each\_id["stock"])  Result[each\_id["category"]] = Sub\_Result  else:  Result[each\_id["category"]]["products"].append(each\_id["name"])  Result[each\_id["category"]]["stock"] += each\_id["stock"]  print(Result) |

1. 請定義一個function，爬取以下網站：<https://tw.stock.yahoo.com/rank/volume?exchange=TAI>  
   並根據名次欄位設計成dict格式回傳

|  |
| --- |
|  |
| # 原答案方法  def get\_rank\_stock():  url = 'https://tw.stock.yahoo.com/rank/volume?exchange=TAI'  response = session.get(url)  title\_line = response.html.xpath("//div[(contains(@class, 'table-header-wrapper'))]") # 標題列  title\_line\_list = title\_line[0].text.split('\n') # 標題列 -> list型式  rows = response.html.xpath("//li[@class='List(n)']") # 每一筆數據資料  data = {}  for each\_row in rows:  Sub\_Result = {}  each\_row\_list = each\_row.text.split('\n') # 每一筆數據資料 -> list型式  for i in range(len(title\_line\_list)):  Sub\_Result.setdefault(title\_line\_list[i], each\_row\_list[i])  del Sub\_Result['名次'] # 名次要當dict.key  data.setdefault(each\_row\_list[0], Sub\_Result)  return data  # 但原答案欄位抓取錯誤，改進後如下  def get\_rank\_stock():  url = 'https://tw.stock.yahoo.com/rank/volume?exchange=TAI'  response = session.get(url)  title\_line = response.html.xpath("//div[(contains(@class, 'table-header-wrapper'))]") # 標題列  title\_line\_list = title\_line[0].text.split('\n') # 標題列 -> list型式  rows = response.html.xpath("//li[@class='List(n)']") # 每一筆數據資料  data = {}  for each\_row in rows:  Sub\_Result = {}  each\_row\_list = each\_row.text.split('\n') # 每一筆數據資料 -> list型式  each\_row\_list[1] = each\_row\_list[1] + ' / ' + each\_row\_list[2] # 修改數據  del each\_row\_list[2]  for i in range(len(title\_line\_list)):  Sub\_Result.setdefault(title\_line\_list[i], each\_row\_list[i])  del Sub\_Result['名次'] # 名次要當dict.key  data.setdefault(each\_row\_list[0], Sub\_Result)  return data |