

ĐỀ BÀI FINAL TEST KHÓA HỌC DATA FOR EVERYONE

Thông tin chung:

Khóa học:	XData_Level 1_Data for Everyone
Điểm tối đa:	10 điểm
Trọng số điểm tốt nghiệp:	50%
Hình thức làm bài:	Làm bài cá nhân, tại nhà
Cấu trúc đề	SQL - 4 điểm, PBI - 3 điểm, Python - 3 điểm
Thời gian giao đề:	Buổi học số 12 khóa học
Thời gian nộp bài:	Trước khi bắt đầu buổi học số 14 của khóa học
Quy định về việc nộp muộn:	Được phép nộp muộn không quá 5h sau deadline Nếu nộp muộn trong khoảng thời gian này, bài làm sẽ bị trừ 1 điểm và quá khoảng thời gian này, bài làm sẽ không được chấp nhận
Quy chế thi chi tiết:	Tai đây
Hướng dẫn nộp bài:	<p>Nén 4 files sau thành một file có đuôi ở dạng zip (vui lòng không nén ở định dạng rar), đặt tên theo cú pháp sau: “Mã lớp_Mã nhóm_Họ và tên_D4E_Final_Test” và nộp tại link</p> <p>Ví dụ: “D4E100_2_Nguyen Ngoc Minh_D4E_Final_Test”</p> <ul style="list-style-type: none">• .SQL chứa tất cả các đoạn mã truy vấn của phần bài SQL• .PBIX chứa file trực quan hoá dữ liệu với PBI• .PDF là file báo cáo phân tích của câu 3 phần PBI• .IPYNB là file python notebook, chứa source code của phần làm bài Python

Dataset và tài liệu liên quan

Bộ dữ liệu / Dataset : [Link](#) - Thông tin về bộ dữ liệu: Data Dictionary

Tên cột	Mô tả
Invoice ID	Mã đơn hàng
Branch	Chi nhánh bán hàng (có 3 chi nhánh A, B ,C)
City	Thành phố đặt chi nhánh bán hàng
Customer Type	Loại khách hàng, có 2 loại khách hàng: Members chỉ những khách hàng có đăng ký thẻ thành viên, Normal chỉ những khách hàng không đăng ký hoặc không có thẻ thành viên
Product line	Loại sản phẩm. Có các loại sản phẩm sau: Electronic accessories, Fashion accessories, Food and beverages, Health and beauty, Home and lifestyle, Sports and travel
Unit Price	Giá bán của 1 sản phẩm
Quantity	Số sản phẩm được mua
Date	Ngày mua hàng
Time	Thời điểm mua hàng (giờ mua hàng)
Payment	Hình thức thanh toán của khách hàng (có 3 loại - Cash, Credit Card, Ewallet)
cogs	Giá vốn bán hàng

Đề bài chi tiết

Lưu ý: Với phần SQL, kết quả cần giống kết quả mẫu. Nếu kết quả không giống kết quả mẫu, học viên không được tính điểm của câu đó.

I. SQL:

Câu 1:

A. Bạn hãy xây dựng đoạn truy vấn đếm tổng số đơn hàng được order (0.25đ)

Kết quả mẫu:

	Total Order ▾
1	1000

B. Bạn hãy xây dựng đoạn truy vấn tính tổng doanh số theo từng Branch. Kết quả làm tròn đến 2 chữ số thập phân (0.25đ)

Kết quả mẫu:

	Branch ▾	Sum Sales ▾
1	A	101143.21
2	C	105303.53
3	B	101140.64

Câu 2:

A. Bạn hãy xây dựng đoạn truy vấn tính tổng doanh số và số lượng đơn hàng của từng Product Line (0.25đ)

Kết quả mẫu:

	Product_line ▾	Sum Sales ▾	Total Order ▾
1	Fashion accessories	51719.9	178
2	Health and beauty	46851.18	152
3	Electronic accessories	51750.03	170
4	Food and beverages	53471.28	174
5	Sports and travel	52497.93	166
6	Home and lifestyle	51297.06	160

B. Bạn hãy xây dựng đoạn truy vấn tính tổng doanh số, tổng số đơn hàng của từng loại khách hàng theo từng Product line. (tổng doanh số làm tròn đến 2 chữ số thập phân) (0.5đ)

Kết quả mẫu:

	Product_line	Customer_type	SumSales	Total Order
1	Electronic accessories	Member	23331.9	78
2	Electronic accessories	Normal	28418.13	92
3	Fashion accessories	Member	25070.44	86
4	Fashion accessories	Normal	26649.46	92
5	Food and beverages	Normal	23606.88	80
6	Food and beverages	Member	29864.4	94
7	Health and beauty	Normal	22250.19	79
8	Health and beauty	Member	24600.99	73
9	Home and lifestyle	Normal	24651.32	77
10	Home and lifestyle	Member	26645.74	83
11	Sports and travel	Normal	25608.12	79
12	Sports and travel	Member	26889.81	87

Câu 3:

A. Với tháng có doanh số cao nhất, bạn hãy tìm ra các khung giờ có tổng số đơn hàng cao hơn số lượng đơn hàng trung bình theo giờ của tháng đó. (0.75đ)

Kết quả mẫu:

	HOUR	Total Order
1	10	37
2	11	36
3	13	40
4	15	33
5	19	35

B. Với mỗi Product line, đều có 2 loại khách hàng (Customer Type) mua hàng là Normal, Member. Bạn hãy tìm các Product line có loại khách hàng mua ít đơn hàng nhưng lại có doanh số cao hơn loại khách hàng còn lại.(1đ)

Kết quả mẫu:

	Product_line ▾	Customer_type ▾	SumSales ▾	Total Order ▾
1	Health and beauty	Member	24600.99	73

Câu 4: Bạn hãy xây dựng đoạn truy vấn tìm ra tổng doanh số, tổng số đơn hàng theo tháng, tổng doanh số và tổng số đơn hàng của các tháng về trước (1đ)

	Month ▾	SumSales ▾	TotalOrder ▾	Total Sales Before ▾	Total OrderBefore ▾
1	1	110754.16	352	NULL	NULL
2	2	92589.88	303	110754.16	352
3	3	104243.34	345	203344.04	655

II. Power BI:

Câu 1: Bạn hãy vẽ biểu đồ để thể hiện các thông tin sau:

- Top 2 City có tổng doanh số lớn nhất (0.5đ)
- Tỷ trọng doanh số theo từng Product line (0.25đ)
- Doanh số của các sản phẩm theo thời gian (0.25đ)

Câu 2:

A. Bạn hãy tạo một calculated column có tên Tax fee để lưu trữ thuế của đơn hàng. Biết rằng Thuế được tính theo 5% của Cogs (0.5đ)

B. Bạn hãy tạo một calculated column có tên Total để lưu trữ tổng số tiền mà khách hàng phải trả cho đơn hàng bao gồm số tiền cho đơn hàng và chi phí thuế (0.5đ)

Câu 3: Dựa vào các thông tin ở phần SQL và PBI, bạn hãy phân tích tình hình kinh doanh của cửa hàng trên? (1đ) !Lưu ý: Bài phân tích cần có các biểu đồ vẽ bằng PBI và các insights rút ra được. Ở định dạng pdf. Tham khảo [link](#) để hiểu rõ hơn cấu trúc bài làm.

III. Python:

Câu 1: Bạn hãy xây dựng một hàm nhận tham số truyền vào là 1 chuỗi ký tự bất kỳ. Hàm kiểm tra và trả về kết quả chuỗi trên là một đoạn password hợp lệ hay không, biết rằng password hợp lệ cần thoả mãn các tiêu chí:

- Chứa 1 ký tự đặc biệt trong các ký tự !, @, #, \$, ^
- Chứa ít nhất một ký tự số
- Chứa ít nhất một ký tự chữ cái viết hoa
- Chứa ít nhất một ký tự chữ cái viết thường
- Độ dài từ 8 - 15 ký tự. (0.75đ)

Câu 2: Bạn hãy đọc [file dữ liệu](#) với pandas. Và giải quyết các câu sau:

- A. Bạn hãy lấy ra các đơn hàng thuộc Branch A hoặc C (0.25đ)
- B. Bạn hãy tìm ra các đơn hàng mua sản phẩm là 'Health and beauty' hoặc 'Sports and travel' (0.25đ)
- C. Cột InvoiceID, giá trị được lưu trữ theo dạng XXX-StoreID-XXXX.

Bạn hãy tạo 1 dataframe mới gồm các cột như trên file và tạo thêm một cột StoreID được tách từ InvoiceID.

Sau đó tính tổng doanh số của từng StoreID đó. (0.25đ)

Câu 3: Bạn hãy xây dựng một hàm nhận tham số truyền vào là một danh sách bất kỳ chỉ chứa các số. Hàm tính toán

và trả về số lớn nhất trong danh sách. Với trường hợp danh sách gồm các số bằng nhau, hàm trả về số lớn nhất. Nếu

có nhiều giá trị bằng số lớn nhất trong danh sách, hàm trả về 1 giá trị duy nhất. (1.5đ)

Ví dụ:

- Với danh sách [1,1,1,1,1,1] -> Hàm trả về 1

- Với danh sách [1,2,1,1,2,3] -> Hàm trả về 2

Good luck!