

TEST 1

USING PYTHON TO REPLICATE EXCEL FUNCTIONS

Trading Service Department

Using given data, do the following tasks:

Dựa vào dữ liệu đã cho, thực thi các tác vụ sau:

Question 1: Write function **TenKH** accepting any valid account number and returning corresponding customer's name

Câu 1: *Viết hàm **TenKH** nhận đầu vào là một số tài khoản bất kỳ và trả ra tên khách hàng tương ứng*

Question 2: Create a list named **fee_for_assessment** which shares the same length of list **taikhoan** so that each element in **fee_for_assessment** follows the following formula:

Câu 2: *Tạo một list tên **fee_for_assessment** có cùng độ dài với list **taikhoan** sao cho mỗi phần tử trong **fee_for_assessment** tuân theo công thức sau*

$$\text{fee_for_assessment}_i = \text{tongphitrungbinh}_i \times 100\% + \text{tonglaitrungbinh}_i \times 30\%$$

Question 3: Write function **TSRBQ_theo_KH** accepting any valid account number or customer's name and returning corresponding **taisanrongbinhquan**

Câu 3: *Viết hàm **TSRBQ_theo_KH** nhận đầu vào là số tài khoản hoặc tên khách hàng bất kỳ và trả ra giá trị **taisanrongbinhquan** tương ứng*

Question 4: Write function **PhiGD_theo_CN** which accepts any valid branch name and returns total **phiGD** yielded by that branch

Câu 4: *Viết hàm **PhiGD_theo_CN** nhận đầu vào là tên chi nhánh bất kỳ và trả ra giá trị tổng số **phiGD** tương ứng*

Question 5: Write function **XepHang_CN** that accepts either "lowest" or "highest" and return the branch that has lowest or highest **phiGD**, accordingly

Câu 5: *Viết hàm **XepHang_CN** nhận đầu vào là một trong hai chuỗi "lowest" hoặc "highest" và trả ra tên chi nhánh có số **phiGD** thấp nhất hoặc cao nhất tương ứng*

Risk Management Department

Using given data, do the following tasks:

Dựa vào dữ liệu đã cho, thực thi các tác vụ sau:

Question 1: Write function **TotalLoans** accepting any valid custody number and returning corresponding customer's outstanding loans

Câu 1: Viết hàm **TotalLoans** nhận đầu vào là một số tài khoản bất kỳ và trả ra dư nợ của khách hàng tương ứng

Question 2: From list **RMR0062** and list **RMR0035**, create a list named **total_cash** so that element i -th in **total_cash** follows the following Excel's fomula:

Câu 2: Từ list **RMR0062** và list **RMR0035**, tạo một list tên **total_cash** sao cho phần tử thứ i trong **total_cash** tuân theo công thức Excel sau:

$$\text{total_cash}_i = \text{IF}(\text{RMR0062}_i - \text{RMR0035}_i > 0, \text{RMR0062}_i, \text{RMR0035}_i)$$

Question 3: From list **total_cash**, **total_loans**, **DL1**, **RMR0015**, create a list named **MMR** so that element i -th in **MMR** follows the following Excel's fomula:

Câu 3: Từ list **total_cash**, **total_loans**, **DL1**, **RMR0015**, tạo một list tên **MMR** sao cho phần tử thứ i trong **MMR** tuân theo công thức Excel sau:

$$\text{MMR}_i = \text{IF}(\text{total_cash}_i - \text{total_loans}_i > 0, 100, \text{IFERROR}((\text{RMR0015}_i + \text{DL1}_i - \text{MAX}(\text{total_loans}_i - \text{total_cash}_i, 0)) / (\text{RMR0015}_i + \text{DL1}_i) * 100, 0))$$

Question 4: Write function **DuNo_theo_CN** which accepts any valid branch name and returns total outstanding loans of that branch

Câu 4: Viết hàm **DuNo_theo_CN** nhận đầu vào là tên chi nhánh bất kỳ và trả ra giá trị tổng dư nợ của chi nhánh đó

Question 5: Write function **XepHang_CN** that accepts either "lowest" or "highest" and return the branch that has lowest or highest **total_loans**, accordingly

Câu 5: Viết hàm **XepHang_CN** nhận đầu vào là một trong hai chuỗi "lowest" hoặc "highest" và trả ra tên chi nhánh có số **total_loans** thấp nhất hoặc cao nhất tương ứng

Brokerage

Using given data, do the following tasks:

Dựa vào dữ liệu đã cho, thực thi các tác vụ sau:

Question 1: Write function **CleanName** accepting a list containing strings and returning a new list of corresponding uppercased strings. Use this function to uppercase list **name** and list **manager**

Câu 1: Viết hàm **CleanName** nhận đầu vào là một list chứa các string bất kỳ và trả ra một list tương ứng chứa các string đã được in hoa. Sử dụng hàm này để in hoa list **name** và list **manager**

Question 2: Write function **Manager_of_Employee** accepting either broker's id or broker's name and returning his/her manager's name

Câu 2: Viết hàm **Manager_of_Employee** nhận đầu vào là số id của broker hoặc tên broker và trả ra tên người quản lý của broker đó

Question 3: Write function **ReachTarget** accepting broker's id and returning either True or False, True if the broker reach target, False otherwise

Câu 3: Viết hàm **ReachTarget** nhận đầu vào là số id của broker và trả ra True hoặc False, True nếu broker đó đạt target, False nếu không đạt target

Question 4: From list **target_fee** and list **fee**, create a new list named **giatritinhbonus** so that i-th element of **giatritinhbonus** follows the following Excel function

Câu 4: Từ list **target_fee** và list **fee**, tạo một list mới tên **giatritinhbonus** sao cho phần tử thứ i của **giatritinhbonus** thỏa công thức Excel sau

$$\text{giatritinhbonus}_i = \text{MAX}(\text{fee}_i - \text{target_fee}_i, 0)$$

Question 5: Write function **TotalFee_by_Manager** that accepts either manager's id or manager's name and returns total fee generated by all employees under his/her management

Câu 5: Viết hàm **TotalFee_by_Manager** nhận đầu vào là số id của manager hoặc tên manager và trả ra tổng phí giao dịch tạo ra bởi tất cả nhân viên được quản lý bởi manager đó.

Human Resource

Using given data, do the following tasks:

Dựa vào dữ liệu đã cho, thực thi các tác vụ sau:

Question 1: Write function **CleanType** accepting a list containing strings and returning a new list of corresponding uppercased strings. Use this function to uppercase list **contract_type**

*Câu 1: Viết hàm **CleanType** nhận đầu vào là một list chứa các string bất kỳ và trả ra một list tương ứng chứa các string đã được in hoa. Sử dụng hàm này để in hoa list **contract_type***

Question 2: Write function **EmployeeInfo** accepting either employee's code or employee's name and returning his/her position and division and gender

*Câu 2: Viết hàm **EmployeeInfo** nhận đầu vào là số mã số nhân viên hoặc tên nhân viên và trả ra chức vụ và phòng ban và giới tính của nhân viên đó*

Question 3: Write function **BackFront** accepting either employee's code or employee's name, this function returns True if the employee is in back office, False if the employee is in front office, "Không xác định" otherwise

*Câu 3: Viết hàm **BackFront** nhận đầu vào là số mã số nhân viên hoặc tên nhân viên, hàm này trả ra True nếu nhân viên đó thuộc back office, False nếu thuộc front office, "Không xác định" cho các trường hợp còn lại*

Question 4: Write function **Days_by_Division** that accepts any division and returns average days at work of that division's employees

*Câu 4: Viết hàm **Days_by_Division** nhận đầu vào là tên một phòng ban bất kỳ và trả ra số ngày làm việc trung bình của nhân viên phòng ban đó*

Question 5: Write function **Days_by_Position_and_Division** that accepts any position and division and returns average days at work of that division's employees whose position is specified

*Câu 5: Viết hàm **Days_by_Position_and_Division** nhận đầu vào là chức vụ và phòng ban và trả ra số ngày làm việc trung bình của nhân viên trong phòng ban có chức vụ đó*

IB02

Using given data, do the following tasks:

Dựa vào dữ liệu đã cho, thực thi các tác vụ sau:

Question 1: Write function **CleanName** accepting a list containing strings and returning a new list of corresponding uppercased strings. Use this function to uppercase list **tenkhachhang**

*Câu 1: Viết hàm **CleanType** nhận đầu vào là một list chứa các string bất kỳ và trả ra một list tương ứng chứa các string đã được in hoa. Sử dụng hàm này để in hoa list **tenkhachhang***

Question 2: From list **khoiluong** and list **gia**, create list **giatrigiaodich** and list **phigiaodich** so that i-th element of **giatrigiaodich** and **phigiaodich** follows the following Excel functions

*Câu 2: Từ list **khoiluong** và list **gia**, tạo hai list mới tên **giatrigiaodich** và **phigiaodich** trong đó phần tử thứ i của **giatrigiaodich** và **phigiaodich** thỏa công thức Excel sau*

$$\begin{aligned} \text{giatrigiaodich}_i &= \text{khoiluong}_i * \text{gia}_i \\ \text{phigiaodich}_i &= \text{giatrigiaodich}_i * 0.15\% \end{aligned}$$

Question 3: Write function **CPGD_theo_KH** accepting either a customer's account number or a customer's name and returning according customer's trading stocks

*Câu 3: Viết hàm **CPGD_theo_KH** nhận đầu vào là số tài khoản khách hàng hoặc tên khách hàng và trả ra mã cổ phiếu có phát sinh giao dịch của khách hàng tương ứng*

Question 4: Write function **CPBan_theo_KH** accepting either a customer's account number or a customer's name and returning all stock he/she sells

*Câu 4: Viết hàm **CPBan_theo_KH** nhận đầu vào là số tài khoản khách hàng hoặc tên khách hàng và trả ra mã cổ phiếu họ bán*

Question 5: Write function **MuaRong_theo_CP** accepting two parameters, first parameter is either a customer's account number or a customer's name, second parameter is a stock ticker. This function returns the net buy(sell) volume of the customer on the specified stock ticker

*Câu 5: Viết hàm **MuaRong_theo_CP** nhận 2 tham số đầu vào, tham số đầu tiên là số tài khoản khách hàng hoặc tên khách hàng, tham số thứ hai là mã cổ phiếu. Hàm này trả ra khối lượng mua ròng (số dương) hoặc bán ròng (số âm) của khách hàng đó trên mã cổ phiếu đã cho*

Internal Control

Using given data, do the following tasks:

Dựa vào dữ liệu đã cho, thực thi các tác vụ sau:

Question 1: Write function **LoaiLenh** that accepts order number then returns both customer's account number and order type

*Câu 1: Viết hàm **LoaiLenh** nhận đầu vào là số hiệu lệnh và trả ra số tài khoản và loại lệnh tương ứng*

Question 2: Write function **GiaTriGD** that accepts order number and returns trading value of that order in which trading value equals trading volume multiplied by stock price

*Câu 2: Viết hàm **GiaTriGD** nhận đầu vào là số hiệu lệnh và trả ra giá trị giao dịch của lệnh đó, trong đó giá trị giao dịch bằng khối lượng giao dịch nhân với giá*

Question 3: Write function **ChungChiHanhNghe** that accepts employee's id number and returns his/her certification. In case of no certification, returns "Nhân viên này chưa có chứng chỉ hành nghề"

*Câu 3: Viết hàm **ChungChiHanhNghe** nhận đầu vào là số id nhân viên và trả ra mã số chứng chỉ hành nghề của nhân viên đó. Trong trường hợp nhân viên này chưa có chứng chỉ hành nghề, trả ra chuỗi "Nhân viên này chưa có chứng chỉ hành nghề"*

Question 4: Create list **nv_khongcochungchi** containing id numbers of all uncertified employees

*Câu 4: Tạo list **nv_khongcochungchi** chứa số id của tất cả nhân viên chưa có chứng chỉ hành nghề*

Question 5: From lists **sohieulenh**, **cchn_nguoiithuchien**, **cchn_nguoiduyet**, create list **lenhvipham** containing order numbers of all violating orders

*Câu 5: Từ list **sohieulenh**, **cchn_nguoiithuchien**, **cchn_nguoiduyet**, tạo list **lenhvipham** chứa toàn bộ số hiệu lệnh của các lệnh vi phạm*