

CÁC TÍNH TOÁN KINH TẾ VÀ TÀI CHÍNH TRONG HỆ THỐNG THÔNG TIN QUẢN LÝ

Trong các hệ thống thông tin quản lý, vấn đề phân tích sự tác động của các yếu tố đến hiệu quả kinh tế của cơ sở sản xuất kinh doanh cũng như các vấn đề dự báo kinh tế có vai trò định hướng rất quan trọng.

Trong chương này chúng ta sẽ xem xét việc sử dụng công cụ Data Analysis trong Excel để giải quyết bài toán này.

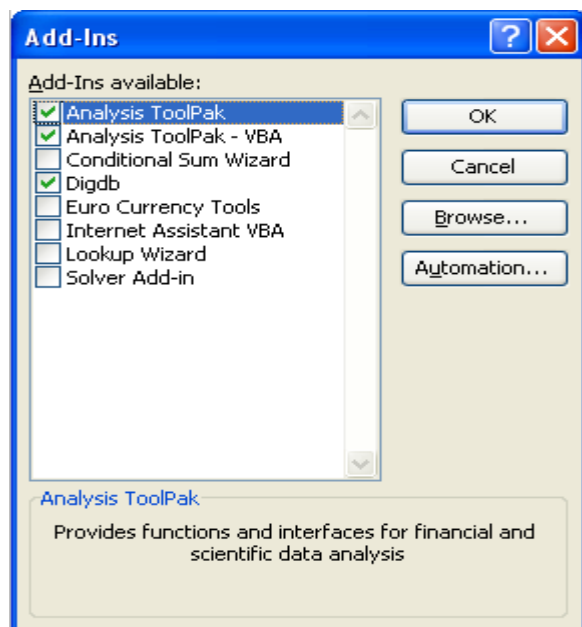
I. Giới thiệu công cụ phân tích Data Analysis.

Trong Excel có một công cụ phân tích rất hiệu quả. Đó là công cụ phân tích Data Analysis.

Các bước làm việc với công cụ Data Analysis như sau:

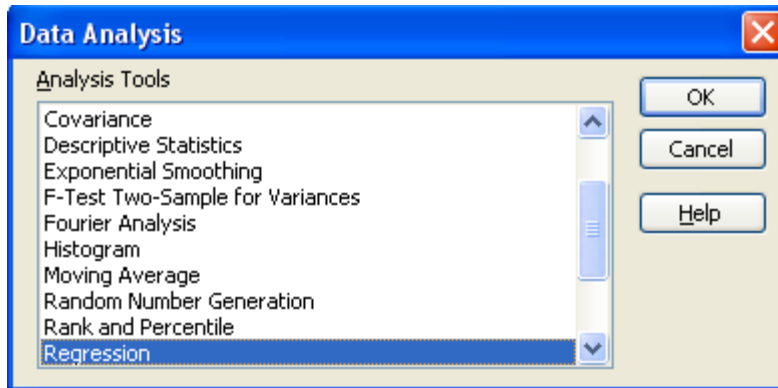
Bước 1: Từ cửa sổ Excel, chọn Tools, chọn Add-Ins...

Xuất hiện cửa sổ như sau:

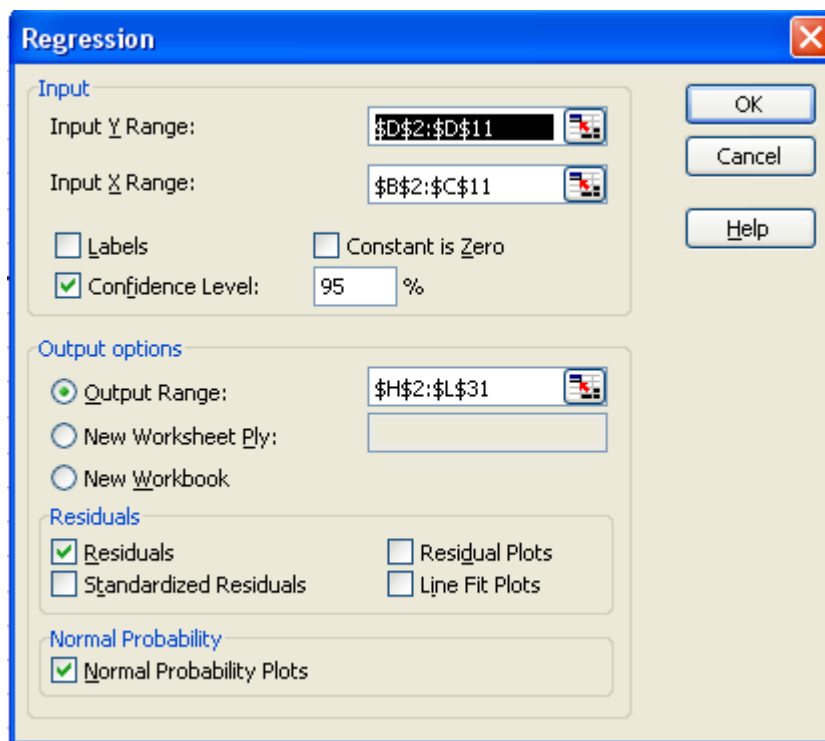


Chọn tích vào Analysis ToolPak và Analysis ToolPak-VBA như trên hình vẽ.

Bước 2: Từ cửa sổ Excel chọn Tools, chọn Data Analysis xuất hiện màn hình sau:



Chọn Regression như trên hình vẽ sẽ xuất hiện màn hình có các tính năng như sau:



Trong Tab Input: Nhập địa chỉ của các ô chứa dữ liệu, nếu chọn Labels xác định hàng đầu tiên không chứa dữ liệu.

Trong Tab Output options:

- Output Range: địa chỉ của các ô chứa kết quả phân tích.
- New Worksheet Ply: chuyển kết quả phân tích đến một bảng tính khác trong cùng Workbook.
- New Workbook: Gửi kết quả phân tích đến một Workbook mới.

Bây giờ chúng ta xem xét việc ứng dụng các công cụ phân tích của Excel để giải quyết bài toán thường gặp trong các hệ thống thông tin quản lý.

II. Phân tích tương quan đơn

Trong các hệ thống thông tin quản lý, chúng ta thường phải nghiên cứu sự ảnh hưởng của các yếu tố nguyên nhân đến yếu tố kết quả. Chẳng hạn, xác định các yếu tố tác động đến năng suất lao động trong doanh nghiệp, các yếu tố ảnh hưởng đến lợi nhuận của một trung tâm thương mại. Chúng ta muốn lượng hóa mối liên hệ này một cách cụ thể hơn bằng một hàm số biểu diễn sự phụ thuộc của yếu tố kết quả vào yếu tố nguyên nhân. Trong trường hợp này, chúng ta phải thiết lập hàm tương quan đơn dưới dạng $Y = AX + B$ trong đó X là yếu tố nguyên nhân còn Y là yếu tố kết quả.

Chúng ta xét một số bài toán sau đây:

Bài toán 1: Trong hệ thống thông tin quản lý sản xuất một doanh nghiệp, để đánh giá sự tác động của đầu tư cho công nghệ mới đến năng suất lao động trong doanh nghiệp, người ta tiến hành thu thập số liệu trong 10 năm. Số liệu được trình bày trong bảng sau:

A Stt	B năm	C NSLĐ	D đầu tư cho công nghệ mới
1	1977	15000	1000
2	1978	15500	1100
3	1979	16000	1250
4	1980	16450	1250
5	1981	16500	1300
6	1982	16700	1350
7	1983	17000	1400
8	1984	18000	1500
9	1985	18500	1550
10	1986	19000	1650

Kết quả phân tích trong Excel như sau:

SUMMARY
OUTPUT

<i>Regression Statistics</i>	
Multiple R	0.984163526
R Square	0.968577846
Adjusted R Square	0.964650077
Standard Error	37.61620476
Observations	10

ANOVA				
	<i>df</i>	<i>SS</i>	<i>MS</i>	<i>F</i>
Regression	1	348930.1691	348930.2	246.5974
Residual	8	11319.83089	1414.979	

Total	9	360250
-------	---	--------

	<i>Coefficients</i>	<i>Standard Error</i>	<i>t Stat</i>	<i>P-value</i>
Intercept	8398.265	544.5869	15.42135	3.1081E-07
X Variable 1	6.342124	0.403869	15.70342	2.6998E-07

Như vậy, mối liên hệ giữa đầu tư cho công nghệ mới và năng xuất lao động trong doanh nghiệp được biểu diễn bằng hàm tương quan đơn có dạng:

$$Y = 6.342124 X + 8398.265$$

Bài toán 2: Cho số liệu về doanh số bán lẻ tháng 12 tháng trong năm 1999 của một trung tâm thương mại và lợi nhuận tương ứng cho trong bảng sau đây:

A	B	C	D
Stt	Tháng	Doanh số bán lẻ	Lợi nhuận
1	1	5000	500
2	2	5100	502
3	3	5210	512
4	4	5200	521
5	5	5400	534
6	6	5545	543
7	7	5630	544
8	8	5700	560
9	9	6000	570
10	10	6100	610
11	11	6300	613
12	12	7000	680

Kết quả phân tích bài toán trong Excel như sau

SUMMARY OUTPUT

<i>Regression Statistics</i>	
Multiple R	0.989536
R Square	0.979182
Adjusted R Square	0.9771
Standard Error	8.128571
Observations	12

ANOVA				
	<i>df</i>	<i>SS</i>	<i>MS</i>	<i>F</i>
Regression	1	31078.18	31078.18	470.3565
Residual	10	660.7367	66.07367	
Total	11	31738.92		

	<i>Coefficients</i>	<i>Standard Error</i>	<i>t Stat</i>	<i>P-value</i>
Intercept	40.50014	23.94977	1.691045	0.121706

X Variable 1	0.090973	0.004195	21.6877	9.71E-10
--------------	-----------------	----------	---------	----------

Như vậy phương trình tương quan biểu diễn sự phụ thuộc của lợi nhuận (Y) vào doanh số (X) như sau:

$$Y = 0.090973 X + 40.50014$$

Phương trình tương quan này có thể sử dụng để dự báo. Giả sử chúng ta muốn dự báo giá trị lợi nhuận khi doanh số đạt mức 8000 triệu đồng. Thay giá trị X = 8000 vào phương trình tương quan ta được:

$$Y = 768.28414$$

III. Phân tích tương quan bội

Phân trên chúng ta đã xét hàm tương quan đơn biểu diễn mối liên hệ của một yếu tố đến yếu tố kết quả. Nhưng trong thực tiễn hoạt động kinh tế, có rất nhiều yếu tố tác động lẫn nhau. Một kết quả trong hoạt động sản xuất kinh doanh là sự tác động tổng hòa của nhiều yếu tố. Yếu tố này tạo tiền đề phát triển cho yếu tố kia. Do đó chúng ta phải xem xét mối liên hệ tương quan giữa nhiều yếu tố với nhau, tức là xét trường hợp tương quan bội.

Bài toán 3: Trong hệ thống Marketing, khi nghiên cứu mức tiêu dùng của dân cư trong một thành phố về mua sắm trang thiết bị điện tử người ta thấy mức tiêu dùng phụ thuộc vào mức thu nhập theo đầu người và tỷ lệ lạm phát, số liệu điều tra cho trong bảng sau đây:

Stt	Thu nhập/đầu người	tỷ lệ lạm phát	Mức tiêu dùng đồ điện
1	2000000	7.56%	200000
2	2500000	7.23%	220000
3	3000000	6.23%	280000
4	3400000	6.12%	320000
5	4000000	5.45%	390000
6	4500000	5.12%	420000
7	5200000	5.02%	510000
8	5500000	4.56%	500000
9	6500000	4.23%	520000
10	7500000	3.98%	540000

Trong trường hợp này hàm tương quan bội có dạng $Y = A_1X_1 + A_2X_2 + B$ trong đó Y là số tiền mua sắm đồ điện, X_1 là thu nhập còn X_2 là tỷ lệ lạm phát.

Kết quả phân tích bằng Data analysis cho kết quả như sau:

SUMMARY OUTPUT

Regression Statistics

Multiple R	0.978249486
R Square	0.956972057
Adjusted R Square	0.944678359
Standard Error	30243.19962
Observations	10

ANOVA				
	<i>df</i>	<i>SS</i>	<i>MS</i>	<i>F</i>
Regression	2	1.42397E+11	7.12E+10	77.84249
Residual	7	6402557864	9.15E+08	
Total	9	1.488E+11		

	<i>Coefficients</i>	<i>Standard Error</i>	<i>t Stat</i>	<i>P-value</i>
Intercept	786183.4198	257676.0143	3.051054	0.018556
X Variable 1	0.014497391	0.020762036	0.698264	0.507538
X Variable 2	-8290394.859	3030777.057	-2.7354	0.029111

Hàm tương quan như sau:

$$Y = 0.014497391X_1 - 8290394.859X_2 + 786183.4198$$

Để dự toán mức tiêu dùng cho đồ điện khi thu nhập đầu người đạt 8000000 và tỷ lệ lạm phát giảm còn 2.5% ta đặt các giá trị $X_1 = 8000\ 000$ và $X_2 = 2.5\%$ và phương trình tương quan ta được kết quả $Y = 694899.53$ đồng.

Bài toán 4: Trong hệ thống thông tin quản lý của một doanh nghiệp, để xác lập mối liên hệ tương quan giữa giá trị tổng sản lượng của doanh nghiệp với các yếu tố đầu tư cho trang thiết bị, đầu tư cho quản lý và đầu tư để nâng cao tay nghề cho công nhân trong một doanh nghiệp chế tạo máy công cụ, người ta thu thập số liệu trong 10 năm và trình bày trong bảng sau đây:

Năm	Giá trị SL	Thiết bị	Quản lý	Tay nghề
1977	50000	500	300	200
1978	52000	500	320	200
1979	53000	510	350	201
1980	54000	555	370	230
1981	55000	560	380	235
1982	55600	600	400	250
1983	57000	700	405	300
1984	59000	750	410	310
1985	61000	800	415	340
1986	62000	850	420	345

Kết quả phân tích như sau:

SUMMARY
OUTPUT

<i>Regression Statistics</i>	
Multiple R	0.989816636
R Square	0.979736973
Adjusted R Square	0.96960546
Standard Error	680.1109209
Observations	10

ANOVA				
	<i>df</i>	<i>SS</i>	<i>MS</i>	<i>F</i>
Regression	3	134188694.8	44729565	96.70194
Residual	6	2775305.188	462550.9	
Total	9	136964000		

	<i>Coefficients</i>	<i>Standard Error</i>	<i>t Stat</i>	<i>P-value</i>
Intercept	29136.85417	3070.045901	9.49069	7.8E-05
X Variable 1	37.86947122	16.80812854	2.253045	0.065168
X Variable 2	38.12997172	12.12191395	3.145541	0.019927
X Variable 3	-44.44386846	41.70611961	-1.06564	0.327577

Như vậy hàm hồi quy bội là:

$$Y = 37.86947122X_1 + 38.12997172 X_2 - 44.44386846X_3$$

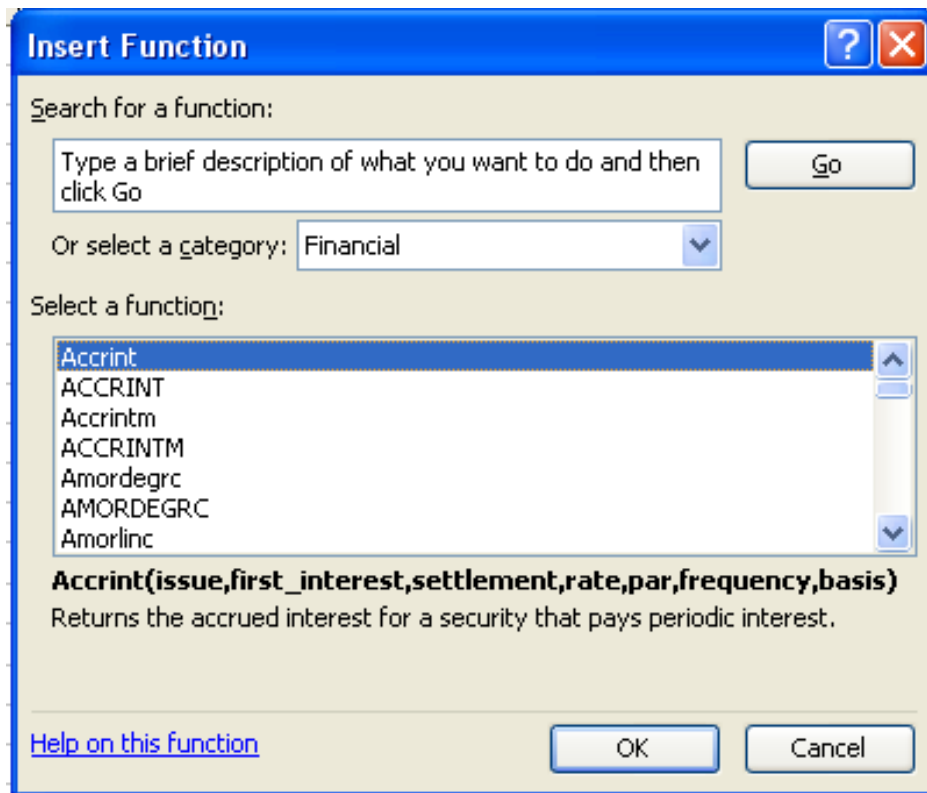
Như vậy giá trị tổng sản lượng của doanh nghiệp sản xuất máy công cụ phụ thuộc rất chặt chẽ vào 3 yếu tố là đầu tư cho trang thiết bị, đầu tư cho công tác quản lý và đầu tư để nâng cao tay nghề của công nhân.

IV. Sử dụng các hàm tài chính trong hệ thống thông tin quản lý.

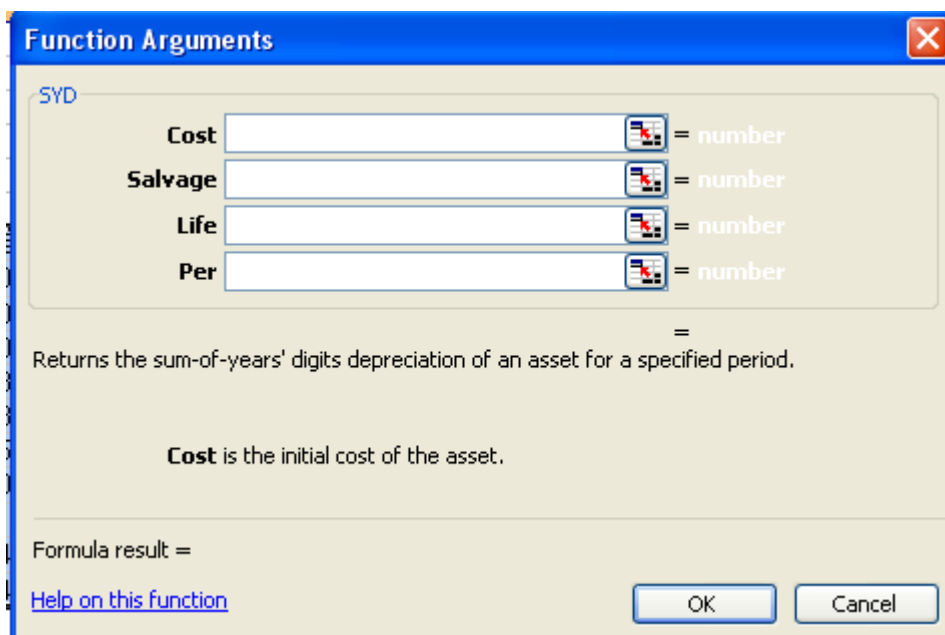
Xem xét việc sử dụng các hàm tài chính của Excel để giải quyết các bài toán khấu hao, tính toán hiệu quả đầu tư.

Trình tự các bước làm việc với hàm trong Excel như sau:

- Chọn lệnh Insert từ Menu lệnh
- Chọn Function
- Trong cửa sổ Or select a category là tên các nhóm hàm mỗi khi ta chọn tên của một nhóm hàm nào đó thì tất cả tên của các hàm trong nhóm được hiện ra trong cửa sổ Select a function để bạn lựa chọn. Ở trong bài này chúng ta chọn Financial xuất hiện bảng sau:



Màn hình giao diện của các hàm tài chính về cơ bản đều bao gồm các mục để người sử dụng nhập địa chỉ của các tham số. Chẳng hạn màn hình của hàm tính toán khấu hao SYD như sau:



Dòng đầu là tên của các hàm và chức năng của nó. Trong mục Formula result = sẽ xuất hiện giá trị của hàm.

Mục Cancel để trở lại bảng tính

Mục Ok để kết thúc. Sau thao tác ngày trong mục Formula result sẽ xuất hiện giá trị của hàm vừa tính toán, đồng thời giá trị này sẽ được đặt vào vị trí nơi có con trỏ trong bảng tính.

V. Tính toán khấu hao tài sản cố định trong hệ thống thông tin quản lý

Tính toán khấu hao tài sản cố định là những vấn đề thường gặp trong quản trị doanh nghiệp. Trong thực tế quản lý người ta thường sử dụng các phương pháp tính toán khấu hao khác nhau. Trong phần này chúng ta xem xét một số hàm tính toán khấu hao cơ bản trong Excel.

1. Hàm SYD

Chức năng: Tính tổng khấu hao hàng năm (Sum of year digits) của tài sản cố định trong một khoảng thời gian.

Quy cách: SYD(Cost, salvage, life, per)

Tham số: Cost là giá trị của tài sản, salvage là giá trị còn lại của tài sản sau khi đã khấu hao, life là đời hữu dụng của tài sản, per là số thứ tự năm khấu hao (năm thứ nhất, năm thứ hai, ...)

Bài toán

Sử dụng hàm SYD tính tổng khấu hao hàng năm cho các tài sản cố định cho trong bảng sau đây (đơn vị tính: triệu đồng)

Tên tài sản	Giá trị ban đầu của tài sản	Giá trị còn lại của tài sản	Số kỳ khấu hao
Máy nổ	150,000,000	5,000,000	4
Máy bào	145,000,000	40,000,000	4
Máy doa	120,000,000	35,000,000	4
Máy phát điện	7,500,000,000	1,500,000,000	4
Bàn máy	1,000,000	100	4
Dây cáp	1,200,000	100	4
Máy đập	35,000,000	45,000,000	4
Máy bàn	120,000,000	30,000,000	4

Các bước tính toán:

- Thiết lập bảng tính từ A₁ đến I₉
- Đưa hộp sáng về ô E₂
- Chọn Function

- Chọn nhóm hàm Financial
- Chọn hàm SYD
- Mục Cost nạp địa chỉ B₂
- Mục Salvage nạp C₂
- Mục life nạp D₂
- Mục Per nạp số 1 (khấu hao năm thứ nhất)
- Tính khấu hao cho các giai đoạn sau cũng tương tự nhưng tham Per nhận các giá trị 2, 3, 4.

Kết quả như sau:

Khấu hao của các kỳ như bảng sau:

kỳ 1	kỳ 2	kỳ 3	kỳ 4	tổng
40,000,000.00	30,000,000.00	20,000,000.00	10,000,000.00	100,000,000.00
42,000,000.00	31,500,000.00	21,000,000.00	10,500,000.00	105,000,000.00
34,000,000.00	25,500,000.00	17,000,000.00	8,500,000.00	85,000,000.00
2,400,000,000.00	1,800,000,000.00	1,200,000,000.00	600,000,000.00	6,000,000,000.00
399,960.00	299,970.00	199,980.00	99,990.00	999,900.00
479,960.00	359,970.00	239,980.00	119,990.00	1,199,900.00
122,000,000.00	91,500,000.00	61,000,000.00	30,500,000.00	305,000,000.00
36,000,000.00	27,000,000.00	18,000,000.00	9,000,000.00	90,000,000.00

Cột tổng là tổng của cả 4 năm khấu hao để so sánh với giá trị ban đầu của tài sản.

2. Tính khấu hao tài sản với tỷ lệ khấu hao cố định

Xét phương pháp tính toán khấu hao khác. Cho biết tỷ lệ khấu hao hàng năm so với giá trị còn lại của tài sản trong năm trước đó. Giá trị còn lại của tài sản trong trường hợp này được tính theo công thức:

$$\text{Salvage} = \text{Cost} * (1 - \text{Rate})^{\text{Life}}$$

Trong đó: Rate là tỷ lệ khấu hao hàng năm.

Bài toán: Tính khấu hao tài sản cố định với các số liệu cho trong bảng sau đây:

Tên tài sản	Giá trị tài sản	Số năm đã sử dụng	Tỷ lệ khấu hao/năm
Máy nổ	15,000,000	6	10%
Máy bào	14,500,000	4	12%
Máy doa	1,200,000	5	15%

Máy phát điện	7,500,000	6	13%
Bàn máy	100,000,000	8	15%
Dây cáp	12,000,000	9	20%
Máy bàn	34,000,000	4	20%

Đơn vị tính: triệu đồng.

- Thiết lập bảng tính từ A₁ đến E₉
- Nạp số liệu vào bảng tính từ dòng 2 đến dòng 9.
- Nạp vào E₂ công thức = B₂*(1-D₂)^C₂
- Sao chép công thức này sang các ô từ E₃ đến E₉.

Kết quả như sau:

TÍNH KHẤU HAO THEO TỶ LỆ KHẤU HAO CỐ ĐỊNH

A	B	C	D	E
Tên tài sản	Giá trị tài sản	Số năm đã sử dụng	Tỷ lệ khấu hao/năm	Giá trị còn lại của tài sản
Máy nổ	15,000,000	6	10%	7971615
Máy bào	14,500,000	4	12%	8695582.72
Máy doa	1,200,000	5	15%	532446.375
Máy phát điện	7,500,000	6	13%	3252196.508
Bàn máy	100,000,000	8	15%	27249052.5
Dây cáp	12,000,000	9	20%	1610612.736
Máy bàn	34,000,000	4	20%	13926400

3. Hàm SLN

Chức năng: Tính khấu hao tài sản với tỷ lệ khấu hao đều trong một khoảng thời gian xác định.

Quy cách: SLN(Cost, salvage, life)

Tham số: Cost: là giá trị ban đầu của tài sản, salvage là giá trị còn lại của tài sản sau khi đã khấu hao, life là đời hữu dụng của tài sản.

Hàm SLN tính khấu hao theo công thức:

$$SLN = (Cost - Salvage)/Life$$

Bài toán

Tính khấu hao theo hàm SLN với các số liệu cho trong bảng sau đây:

Tên tài sản	Giá trị ban đầu	Giá trị còn lại của tài sản	Số kỳ khấu hao
Máy nổ	150,000,000	50,000,000	10
Máy bào	145,000,000	40,000,000	10

Máy doa	120,000,000	35,000,000	5
Máy phát điện	7,500,000,000	1,500,000,000	20
Bàn máy	1,000,000	100	2
Dây cáp	1,200,000	100	2
Máy hàn	120,000,000	30,000,000	4

Đơn vị tính: triệu đồng

Các bước tính toán

Thiết lập bảng tính từ A1 đến F9

- Đưa hộp sáng vào ô E2
- Chọn Insert
- Chọn Function
- Chọn nhóm hàm Financial
- Chọn hàm SLN
- Mục Cost nạp địa chỉ B2.
- Mục Salvage nạp địa chỉ C2
- Mục Life nạp địa chỉ D2

Kết quả như sau:

A	B	C	D	E
Tên tài sản	Giá trị ban đầu	Giá trị còn lại của tài sản	Số kỳ khấu hao	giá trị khấu hao mỗi kỳ
Máy nổ	150,000,000	50,000,000	10	10,000,000.00
Máy bào	145,000,000	40,000,000	10	10,500,000.00
Máy doa	120,000,000	35,000,000	5	17,000,000.00
Máy phát điện	7,500,000,000	1,500,000,000	20	300,000,000.00
Bàn máy	1,000,000	100	2	499,950.00
Dây cáp	1,200,000	100	2	599,950.00
Máy hàn	120,000,000	30,000,000	4	22,500,000.00

VI. Tính toán hiệu quả vốn đầu tư trong các hệ thống thông tin quản lý

Quản lý vốn đầu tư là một trong các vấn đề quan trọng trong hệ thống thông tin tài chính kế toán. Bây giờ, chúng ta xem xét việc sử dụng các hàm tài chính của Excel để giải quyết bài toán tính toán hiệu quả vốn đầu tư của doanh nghiệp

1. Tính lãi gộp

Trong trường hợp nhà đầu tư không rút lãi trong suốt giai đoạn đầu tư thì giá trị tương lai tính theo công thức lãi gộp:

$$FV = PV(1 + \text{Rate})^{\text{Life}}$$

Trong đó: FV là giá trị tương lai của một khoản đầu tư, PV là giá trị ban đầu, Rate là lãi suất hàng năm còn Life là thời gian đầu tư

Bài toán. Tính lãi gộp cho các khoản đầu tư của một doanh nghiệp với các số liệu cho trong bảng sau đây (Đơn vị tính: USD)

Mã hiệu đầu tư	Giá trị hiện tại	Lãi suất	Số năm đầu tư
A1	15,000,000	0.015	5
A2	15,000,000	0.015	5
A3	45,000,000	0.025	10
A4	15,000,000	0.016	12
A5	100,000,000	0.021	15
A6	120,000,000	0.022	10
A7	5,000,000	0.012	5
A8	10,000,000	0.021	12
A9	12,000,000	0.025	12
A10	50,000,000	0.023	15

Các bước tính toán

- Thiết lập bảng tính từ A1 đến E11
- Đưa hộp sáng về vị trí E2
- Nhập công thức $=B2*(1+C2)^{D2}$
- Sao chép công thức trong ô E2 sang các ô từ E3 đến E11

Kết quả như sau

A	B	C	D	E
Mã hiệu đầu tư	Giá trị hiện tại	Lãi suất	Số năm đầu tư	Lãi gộp
A1	15,000,000	0.015	5	16159260
A2	15,000,000	0.015	5	16159260
A3	45,000,000	0.025	10	57603804
A4	15,000,000	0.016	12	18147456
A5	100,000,000	0.021	15	136579693
A6	120,000,000	0.022	10	149172993
A7	5,000,000	0.012	5	5307286.9
A8	10,000,000	0.021	12	12832430
A9	12,000,000	0.025	12	16138666
A10	50,000,000	0.023	15	70324153

2. Hàm FVCHEDULE

Chức năng: Tính giá trị tương lai của vốn đầu tư với dãy lãi suất (SCHEDULE) thay đổi.

Qui cách: FVCHEDULE (principal,schedule)

Tham số: Principal là giá trị hiện tại của một khoản đầu tư, Schedule là một dãy tỷ lệ lãi suất được áp dụng. Hàm FVCHEDULE tính toán theo công thức:

$$FVCHEDULE = Principal * (1 + Rate1) * (1 + Rate2) * \dots * (1 + Rate_n)$$

Trong đó Rate i là lãi suất kỳ thứ i

Bài toán. Tính giá trị tương lai của các khoản đầu tư cho trong bảng sau với lãi suất thay đổi theo từng kỳ bằng hàm FVCHEDULE

Mã hiệu đầu tư	Giá trị ban đầu	Lãi suất kỳ 1	Lãi suất kỳ 2	Lãi suất kỳ 3	Lãi suất kỳ 4
A1	1,000	0.01	0.02	0.01	0.02
A2	1,500	0.01	0.02	0.01	0.02
A3	2,000	0.01	0.02	0.01	0.02
A4	25,400	0.01	0.02	0.01	0.02
A5	3,000	0.01	0.02	0.01	0.02
A6	5,000	0.01	0.02	0.01	0.02
A7	12,000	0.01	0.02	0.01	0.02
A8	15,000	0.01	0.02	0.01	0.02
A9	12,000	0.01	0.02	0.01	0.02
A10	1,000	0.01	0.02	0.01	0.02

Đơn vị tính: USD

Kết quả như sau:

Tính tiền đầu tư với lãi suất thay đổi

A Mã hiệu đầu tư	B Giá trị ban đầu	C Lãi suất kỳ 1	D Lãi suất kỳ 2	E Lãi suất kỳ 3	F Lãi suất kỳ 4	G Tiền lĩnh
A1	1,000	0.01	0.02	0.01	0.02	\$1,061
A2	1,500	0.01	0.02	0.01	0.02	\$1,592
A3	2,000	0.01	0.02	0.01	0.02	\$2,123
A4	25,400	0.01	0.02	0.01	0.02	\$26,957
A5	3,000	0.01	0.02	0.01	0.02	\$3,184
A6	5,000	0.01	0.02	0.01	0.02	\$5,307
A7	12,000	0.01	0.02	0.01	0.02	\$12,736

A8	15,000	0.01	0.02	0.01	0.02	\$15,920
A9	12,000	0.01	0.02	0.01	0.02	\$12,736
A10	1,000	0.01	0.02	0.01	0.02	\$1,061

Đơn vị tính: USD

3. Hàm EFFECT

Chức năng: Tính lãi suất thực tế hàng năm cho một nguồn vốn đầu tư

Qui cách: EFFECT(Nominal_Rate, npery)

Tham số: Nominal_Rate là lãi suất danh nghĩa. Npery là số lần tính lãi trong năm.

Bài toán. Tính lãi suất thực tế hàng năm cho các khoản đầu tư cho trong bảng sau hàm EFFECT

Mã hiệu đầu tư	Lãi suất danh nghĩa	Số lần tính lãi
A1	6025%	4
A2	6.45%	4
A3	7.45%	4
A4	10.15%	2
A5	11.25%	2
A6	9.45%	3
A7	7.65%	3
A8	8.75%	3
A9	8.95%	4
A10	7.95%	4

Kết quả như sau:

A	B	C	D
Mã hiệu đầu tư	Lãi suất danh nghĩa	Số lần tính lãi	Lãi suất thực tế
A1	6.25%	4	6.40%
A2	6.45%	4	6.61%
A3	7.45%	4	7.66%
A4	10.15%	2	10.41%
A5	11.25%	2	11.57%
A6	9.45%	3	9.75%
A7	7.65%	3	7.85%
A8	8.75%	3	9.01%
A9	8.95%	4	9.25%
A10	7.95%	4	8.19%

4. Hàm NPV

Chức năng: Tính giá trị hiện tại ròng của một dự án đầu tư (giá trị hiện tại ròng là số chênh lệch giữa giá trị của các luồng tiền kỳ vọng trong tương lai với giá trị hiện tại của vốn đầu tư)

Qui cách: NPV(Rate, Value1, Value2...)

Tham số: Rate là lãi suất hàng năm. Value1 là giá trị của vốn đầu tư ban đầu (biểu diễn dưới dạng số âm), Value2, value3... là luồng tiền kỳ vọng trong tương lai.

Bài toán. Tính giá trị hiện tại ròng của các dự án đầu tư cho trong bảng sau đây bằng hàm NPV

Mã hiệu đầu tư	Lãi suất	giá trị ban đầu	Năm thứ 1	Năm thứ 2	Năm thứ 3	Năm thứ 4	Năm thứ 5
A1	6.25%	-5,000,000	500,000	600,000	650,000	700,000	800,000
A2	6.45%	-8,000,000	500,000	650,000	700,000	800,000	1,000,000
A3	7.45%	-9,000,000	400,000	500,000	600,000	750,000	800,000
A4	10.15%	-12,500,000	1,000,000	1,200,000	1,300,000	1,500,000	1,600,000
A5	11.25%	-6,500,000	400,000	450,000	500,000	600,000	650,000
A6	9.45%	-12,000,000	1,000,000	1,100,000	1,200,000	1,300,000	1,400,000
A8	8.75%	-15,000,000	2,500,000	2,600,000	2,800,000	2,900,000	3,000,000
A9	8.95%	-25,000,000	4,500,000	4,600,000	4,700,000	4,800,000	5,000,000
A10	7.95%	-10,000,000	1,200,000	1,300,000	1,400,000	1,450,000	1,500,000

Các bước tính toán:

- Thiết lập bảng tính từ A1 đến H11
- Đưa hộp sáng về ô I2
- Chọn Insert
- Chọn function
- Chọn nhóm Financial
- Chọn hàm NPV
- Mục Rate nạp địa chỉ B2
- Value1 nạp địa chỉ C2
- Value2 nạp địa chỉ D2 ...
- Chọn Copy hàm trong ô I2 sang các ô từ I3 đến I11

Kết quả như sau:

Tính giá trị hiện tại ròng của các dự án đầu tư bằng hàm NPV

A B C D E F

Mã hiệu đầu tư	Lãi suất	giá trị ban đầu	năm thứ 1	năm thứ 2	năm thứ 3
A1	6.25%	-5,000,000	500,000	600,000	650,000
A2	6.45%	-8,000,000	500,000	650,000	700,000
A3	7.45%	-9,000,000	400,000	500,000	600,000
A4	10.15%	-12,500,000	1,000,000	1,200,000	1,300,000
A5	11.25%	-6,500,000	400,000	450,000	500,000
A6	9.45%	-12,000,000	1,000,000	1,100,000	1,200,000
A8	8.75%	-15,000,000	2,500,000	2,600,000	2,800,000
A9	8.95%	-25,000,000	4,500,000	4,600,000	4,700,000
A10	7.95%	-10,000,000	1,200,000	1,300,000	1,400,000

G năm thứ 4	H năm thứ 5	I Giá trị hiện tại ròng
700,000	800,000	(\$2,179,710.74)
800,000	1,000,000	(\$4,717,466.73)
750,000	800,000	(\$6,132,920.23)
1,500,000	1,600,000	(\$6,922,112.86)
600,000	650,000	(\$4,171,322.25)
1,300,000	1,400,000	(\$6,811,878.08)
2,900,000	3,000,000	(\$3,935,586.83)
4,800,000	5,000,000	(\$6,146,225.91)
1,450,000	1,500,000	(\$4,232,404.33)

5. Hàm IRR

Chức năng: Tính tỷ lệ nội hoàn của một dự án đầu tư (tỷ lệ nội hoàn vốn nội bộ xác định tỷ lệ hoàn vốn của một dự án đầu tư)

Qui cách: IRR(Value,Guess)

Tham số: Value là giá trị của vốn đầu tư ban đầu (biểu diễn dưới dạng số âm), Guess là tỷ lệ kỳ vọng.

Hàm IRR của một dự án là tỷ lệ chiết khấu mà tại đó NPV = 0

Bài toán. Tính tỷ lệ nội hoàn của các dự án đầu tư cho trong bảng sau bằng hàm IRR

Mã hiệu đầu tư	giá trị ban đầu	năm thứ 1	năm thứ 2	năm thứ 3	năm thứ 4	năm thứ 5	IRR
A1	-5,000,000	1,200,000	1200000	1500000	1500000	1500000	11%
A2	-8,000,000	1,500,000	2,000,000	2,000,000	3,000,000	3,000,000	12%
A3	-9,000,000	1,500,000	2000000	2000000	3000000	5000000	12%
A4	-12,500,000	2,000,000	3000000	3000000	4000000	6000000	11%

A5	-6,500,000	1,200,000	1500000%	2000000	2000000	2000000	0%
A6	-12,000,000	2,000,000	2000000	3000000	4000000	5000000	9%
A8	-15,000,000	3,000,000	3000000	4000000	4500000	5000000	9%
A9	-25,000,000	5,000,000	6000000	6000000	7000000	8000000	8%
A10	-10,000,000	2,000,000	2000000	3000000	3000000	3500000	10%

Chương 10

SỬ DỤNG CƠ SỞ DỮ LIỆU TRONG HỆ THỐNG THÔNG TIN QUẢN LÝ

Như đã nói trong các phần trước đây, trong các hệ thống thông tin quản lý người ta thường sử dụng một số hệ quản trị cơ sở dữ liệu rất thông dụng như Foxpro, Access, Oracle, SQL... Trong chương này chúng ta xem xét việc sử dụng hệ quản trị cơ sở dữ liệu SQL trong các hệ thống thông tin quản lý. Các chương trình được trình bày dưới dạng cơ bản nhất để tiện lợi cho việc theo dõi của học viên.

Ví dụ: Cho CSDL như sau

EMPOYEE

EmpNo	Name	Job	Salary	Comm	DeptNo	Sex
100	Smit	Anlt	3.500		30	M
101	Reed	Mngr	4.500		30	F
102	N Anh	Mngr	3.800		40	F
103	McDonnel	Mngr	2.900		20	M
109	Chen	Spvr	3.600		60	M
110	Allen	Sism	2.700	8.000	40	M
114	Di Salvo	Mngr	4.200		50	F
116	Watson	Sism	5.300		30	F
120	Turner	Anlt	3.700	5.200	20	M

Department

Deptno	Deptname	Loc	Mgr	Exp_budg	Rev_budg
10	Kế toán	Hà nội	200	10.000	
20	Marketing	Hà nội	101	240.000	
30	Nghiên cứu	Hà tây	109	125.000	
40	HỖ trợ khách hàng	Hà nội	210	280.000	800.000
50	Sản xuất	Hà tây	215	130.000	
60	Thành phẩm	Hà tây	104	90.000	

EMPLHIST

EmpNo	Seq	Date_Beg	Date_End	Salary	FrJob	ToJob	Promo	FrDept	ToDept
100	1	10/01/80		1700	Clrk	Clrk	N		10
103	1	01/01/81		3500	Anlt	Anlt	N		30
105	1	01/01/81		4500	Mngr	Mngr	N		30
110	1	01/01/81		1800	Clrk	Clrk	N		50
200	1	01/01/81		2900	Mngr	Mngr	N		10
210	1	01/01/81		3600	Mngr	Mngr	N		50
213	1	01/01/81		1625	Clrk	Clrk	N		60

214	1	01/01/81		825	Drvr	Drvr	N		50
215	1	01/01/81		2700	Spvr	Spvr	N		60

1. Nạp dữ liệu cho một cơ sở dữ liệu

i. Bổ sung bộ giá trị mới

Có hai cách bổ sung bộ giá trị mới cho bảng, cách 1, bổ sung trực tiếp một bộ bởi một lệnh SQL và cách 2, bổ sung nhiều bộ giá trị lấy từ (các) bộ giá trị của các bảng của CSDL.

a). Bổ sung trực tiếp một bộ giá trị.

Cú pháp:

Insert Into <tên bảng> [(<tên cột 1>, <tên cột 2>, ...)]

Values (<biểu thức 1>, <biểu thức 2>, ...);

Ngữ nghĩa: Thêm một bộ giá trị (bản ghi) mới vào bảng có tên được chỉ ra sau từ khóa Into với giá trị của <biểu thức 1> được gán cho <tên cột 1>, <biểu thức 2> được gán cho <tên cột 2> v.v...

Lưu ý: số lượng biểu thức và kiểu giá trị của các biểu thức phải tương ứng với số lượng và kiểu giá trị của các tên cột trong danh sách tên.

Ví dụ 1: Thêm một phòng mới có tên Chức năng, mã 70 đặt tại địa điểm Hải phòng, kinh phí hoạt động là 250000\$/năm và phòng không có doanh thu và chưa có người phụ trách.

Insert Into Department

(Deptno, Deptname, Loc, Mgr, Exp_budg, Rev_budg)

Values(70, 'Chức năng', 'Hải Phòng', Null, 250000, Null);

Kết quả ta có bảng Department với kết quả như sau:

Deptno	Deptname	Loc	Mgr	Exp_budg	Rev_budg
10	Kế toán	Hà nội	200	10.000	
20	Marketing	Hà nội	101	240.000	
30	Nghiên cứu	Hà tây	109	125.000	
40	HỖ trợ khách hàng	Hà nội	210	280.000	800.000
50	Sản xuất	Hà tây	215	130.000	
60	Thành phẩm	Hà tây	104	90.000	
70	Chức năng	Hải phòng		250.000	

b). Thêm một hay nhiều bộ giá trị từ bảng CSDL

Cú pháp:

```

Insert Into<tên bảng>[(<tên cột 1>,<tên cột 2>...)]
Select <biểu thức 1>, <biểu thức 2>,...
From<danh sách các bảng nguồn>
[Where<điều kiện>]
[Group By <danh sách cột phân nhóm>]
[Order By <cột 1>[ASC|DESC], <cột 1>[ASC|DESC], ...]
[Having<điều kiện>];

```

Ngữ nghĩa: Cũng như trên, số lượng biểu thức và kiểu giá trị của các biểu thức sau Select phải phù hợp với số lượng và kiểu của các cột có tên trong danh sách đi sau tên bảng, đồng thời phải phù hợp với các ràng buộc toàn vẹn được định nghĩa trên quan hệ đó. Nếu bộ giá trị Select được vi phạm ràng buộc toàn vẹn định nghĩa trên quan hệ được bổ sung thì sẽ có các thông báo lỗi thích hợp và bộ đó không được bổ sung vào bảng.

Ví dụ: Bổ sung các bản ghi cho bảng EMPLHIST đối với những nhân viên chưa có quá trình công tác và trong bảng, với giả thiết thêm rằng họ được tuyển dụng vào lại tại công ty kể từ ngày 01/01/1980. Câu lệnh được SQL được viết như sau:

Insert into EMPLHIST

EmpNo	Seq	Date_Beg	Date_End	Salary	FrJob	ToJob	Promo	FrDept	ToDept
100	1	10/01/80		1700	Clrk	Clrk	N		10
103	1	01/01/81		3500	Anlt	Anlt	N		30
105	1	01/01/81		4500	Mngr	Mngr	N		30
110	1	01/01/81		1800	Clrk	Clrk	N		50
200	1	01/01/81		2900	Mngr	Mngr	N		10
210	1	01/01/81		3600	Mngr	Mngr	N		50
213	1	01/01/81		1625	Clrk	Clrk	N		60
214	1	01/01/81		825	Drvr	Drvr	N		50
215	1	01/01/81		2700	Spvr	Spvr	N		60

ii. Tạo mới một bảng với các bộ giá trị lấy từ CSDL

Các câu truy vấn dữ liệu để tìm kiếm thông tin tạo ra một bảng trung gian với những mối liên hệ sao cho có thể xem và, nếu được phép, có thể sửa chữa dữ liệu hoặc xóa bỏ chúng.

Cú pháp:

```
Select<biểu thức 1>, <biểu thức 2>, ...
```

From < danh sách các bảng nguồn >
 Into Table < tên bảng >
 [Where < điều kiện >]
 [Group By < danh sách cột phân nhóm >]
 [Order By < cột 1 > [ASC|DESC]], < cột 1 > [ASC|DESC], ...]
 [Having < điều kiện >];

Ví dụ: Tạo bảng mới tên là Manager bao gồm chỉ những nhân viên phụ trách các phòng ban.

```
Select* From Employee Into Manager
Where Job='Mngr' Order By DeptNo;
```

Kết quả ta có một bảng mới Manager với các bản ghi sau:

EmpNo	Name	Job	Salary	Comm	DeptNo	Sex
101	Reed	Mngr	4.500		30	F
102	N Anh	Mngr	3.800		40	F
103	McDonnel	Mngr	2.900		20	M
114	Di Salvo	Mngr	4.200		50	F

iii. Sửa nội dung bản ghi

Thông thường có thể sửa nội dung của các bản ghi bằng cách cho hiển thị nội dung của bảng, di chuyển con trỏ đến bản ghi cần sửa và thực hiện việc thay đổi, nhưng đó chỉ là các bản ghi nhỏ đối với các bản ghi lớn thì việc tìm vị trí đó là khó khăn vậy cung cấp một lệnh cho phép sửa đổi nội dung các bản ghi trong CSDL một cách dễ dàng, chính xác và nhanh chóng.

Cú pháp:

```
Update < tên bảng >
Set < tên cột 1 > = < biểu thức 1 >,
< tên cột 2 > = < biểu thức 2 >, ...
< tên cột n > = < biểu thức n >
```

[Where < điều kiện >];

Ngữ nghĩa: Giá trị của các trường có tên trong danh sách < tên cột 1 >, < tên cột 2 >, ... của những bản ghi thỏa mãn điều kiện sau where sẽ được sửa đổi thành giá trị của các < biểu thức 1 >, < biểu thức 2 >, ... tương ứng. Nếu không có mệnh đề where, thì tất cả các bản ghi của bảng sẽ được sửa đổi.

Ví dụ: Tăng lương thêm 10% cho các nhân viên phụ trách các phòng ban

```
Update Employee
Set Salary=Salary*1.1
Where Job='Mngr'
```

Kết quả là, lương mới của các nhân viên phụ trách các phòng ban được thể hiện như trong bảng dưới đây.

EmpNo	Name	Job	Salary	Comm	DeptNo	Sex
101	Reed	Mngr	4.950		30	F
102	N Anh	Mngr	4.180		40	F
103	McDonnel	Mngr	3.190		20	M
114	Di Salvo	Mngr	4.620		50	F

iv. Xóa bản ghi khỏi bảng

Việc loại bỏ một bản ghi khỏi một bảng trong CSDL là một trong những thao tác cập nhật dữ liệu được tiến hành một cách thường xuyên nhằm đảm bảo phản ánh tình trạng mới nhất của CSDL.

Cú pháp:

```
Delete From<tên bảng>
```

```
[Where<điều kiện>];
```

Ngữ nghĩa: Các bản ghi thỏa điều kiện sau Where sẽ bị xóa khỏi bảng. Nếu không có mệnh đề Where thì tất cả các bản ghi của bảng sẽ bị xóa khỏi bảng.

Ví dụ: Xóa tất cả các nhân viên phụ trách các phòng ban khỏi bảng nhân viên Employee.

```
Delete From Employee
```

```
Where Job='Mngr'
```

Kết quả là, bảng Employee chỉ còn các bản ghi sau:

EmpNo	Name	Job	Salary	Comm	DeptNo	Sex
100	Smit	Anlt	3.500		30	M
109	Chen	Spvr	3.600		60	M
110	Allen	Sism	2.700	8.000	40	M
116	Watson	Sism	5.300		30	F
120	Turner	Anlt	3.700	5.200	20	M

2. Tìm kiếm trong cơ sở dữ liệu

Câu lệnh Select –SQL tìm kiếm dữ liệu là một trong số các câu lệnh SQL cài đặt đầy đủ các phép toán quan hệ dựa trên các từ khóa cơ bản Select, From, Where,

Group By, Order By, Having. Đây là câu lệnh được sử dụng phổ biến nhất với mục đích tìm kiếm thông tin trong CSDL quan hệ. Cú pháp tổng quát của câu lệnh như sau:

```
Select[Distinct]<biểu thức 1>, <biểu thức 2>, ...
From <tên bảng 1>, <tên bảng 2>, ...
[Where <điều kiện chọn>]
[Group By <tên cột 1>, <tên cột 2>, ...]
[Order By <tên cột 1>|<biểu thức số 1>[ASC|DESC], ...]
Having<điều kiện in kết quả>;
```

i. Tìm thông tin từ các cột của bảng

```
Select{*|<biểu thức 1> [AS<tên mới 1>],
<biểu thức 2>[AS<tên mới 2>], ...}
From<tên bảng>;
```

Ví dụ:

Khi cần lấy thông tin về tất cả các cột của bảng chúng ta có thể sử dụng dấu (*) thay cho việc liệt kê các tên cột của bảng. Câu lệnh trên tương đương với câu lệnh:

```
Select * From Department;
```

Kết quả của câu lệnh là một bảng:

Deptno	Deptname	Loc	Mgr	Exp_budg	Rev_budg
10	Kế toán	Hà nội	200	10.000	
20	Marketing	Hà nội	101	240.000	
30	Nghiên cứu	Hà tây	109	125.000	
40	HỖ trợ khách hàng	Hà nội	210	280.000	800.000
50	Sản xuất	Hà tây	215	130.000	
60	Thành phẩm	Hà tây	104	90.000	

Ví dụ: Cho mã số, tên địa điểm, kinh phí hoạt động của từng phòng ban trong công ty:

```
Select DeptNo, DeptName, Loc, Exp_Budg
From Department;
```

Kết quả là bảng như sau:

Deptno	Deptname	Loc	Exp_budg
10	Kế toán	Hà nội	10.000
20	Marketing	Hà nội	240.000

30	Nghiên cứu	Hà tây	125.000
40	HỖ trợ khách hàng	Hà nội	280.000
50	Sản xuất	Hà tây	130.000
60	Thành phẩm	Hà tây	90.000

Giả thiết cần đặt tên cho các cột thay vì hiển thị bằng tiếng anh câu lệnh được viết như sau:

```
Select DeptNo As[Mã số], DeptName As [Tên phòng], Loc As [Địa điểm],
Exp_budg As [Kinh phí]
```

From Department;

Kết quả của câu lệnh như sau:

Mã số	Tên phòng	Địa điểm	Kinh phí
10	Kế toán	Hà nội	10.000
20	Marketing	Hà nội	240.000
30	Nghiên cứu	Hà tây	125.000
40	HỖ trợ khách hàng	Hà nội	280.000
50	Sản xuất	Hà tây	130.000
60	Thành phẩm	Hà tây	90.000

ii. Chọn các dòng của bảng- mệnh đề Where

Nhiều trường hợp chúng ta chỉ cần chọn ra những bộ giá trị của bảng thỏa mãn điều kiện nào đó. Mệnh đề Where với cú pháp Where<điều kiện> cho phép thực hiện điều đó. ở đây <điều kiện> là một biểu thức mà kết quả là một giá trị logic đúng (True) hoặc sai (False). Đây là sự cài đặt của phép chọn trong đại số quan hệ.

Ví dụ:

Cho danh sách nhân viên phòng số 40?:

```
Select * From Employee Where Deptno=40;
```

Kết quả là bảng sau:

EmpNo	Name	Job	Salary	Comm	DeptNo	Sex
102	N Anh	Mngr	3.800		40	F
110	Allen	Sism	2.700	8.000	40	M

Ví dụ:

Cho danh sách nhân viên phòng số 10, 30, 50

```
Select* From Employee
```

```
Where (deptNo=10) OR (deptNo=30) or (deptno = 50)
```

EmpNo	Name	Job	Salary	Comm	DeptNo	Sex
100	Smit	Anlt	3.500		30	M

101	Reed	Mngr	4.500		30	F
114	Di Salvo	Mngr	4.200		50	F
116	Watson	Slsm	5.300		30	F

iii. Thứ tự hiển thị các bản ghi – Mệnh đề Order by

Để thực hiện sắp xếp nhân viên, SQL hỗ trợ bởi mệnh đề Order By để sắp xếp kết quả tìm được.

Cú pháp:

Order By <tên cột>|<biểu thức>[ASC|DESC],<tên cột>|<biểu thức>[ASC|DESC], ...

Biểu thức phải có giá trị số.

Ví dụ: cho danh sách các nhân viên của phòng 10, 30, 50. Kết quả in ra theo thứ tự tăng dần của mã phòng và giảm dần theo mức lương

```
Select*From Employee where Deptno In (10,30,50)
```

```
Order By 6 ASC, 4 DESC;
```

Kết quả bảng:

iv. Điều kiện hiển thị các bản ghi – mệnh đề Having

Mệnh đề Where cho phép chọn các bản ghi của bảng thỏa mãn điều kiện tìm kiếm. Trong một số trường hợp sau khi tìm được các bản ghi thỏa điều kiện tìm, chúng ta chỉ muốn hiển thị chỉ những bản ghi thỏa một điều kiện khác nữa. SQL hỗ trợ yêu cầu này bởi mệnh đề Having<điều kiện>. Thông thường mệnh đề này được áp dụng trong những câu lệnh tìm các bộ giá trị thông qua các quá trình tính toán trên nhóm

Ví dụ:liệt kê danh sách các nhân viên phòng 10,30,50. Chỉ in những người là lãnh đạo phòng ban.

```
Select *From Employee Where Deptno IN (10,30,50)
```

```
Order By 6 ASC, 4DESC
```

```
Having Job = “Mngr”;
```

Kết quả là bảng:

EmpNo	Name	Job	Salary	Comm	DeptNo	Sex
101	Reed	Mngr	4.500		30	F
114	Di Salvo	Mngr	4.200		50	F

v. Truy vấn thông tin từ nhiều bảng dữ liệu

Bản chất là giống như trên một bảng, tức là cần chỉ ra thông tin gì cần tìm và lấy từ các nguồn dữ liệu nào. Các bảng dữ liệu nguồn này cần chỉ ra trong mệnh đề From trong câu lệnh Select.

Nếu các bảng dữ liệu nguồn có các tên thuộc tính giống nhau thì tên thuộc tính này phải được viết tường minh trong biểu thức tìm kiếm với tên bảng đi kèm phía trước. Nói chung trong một CSDL quan hệ, các bảng thường có các mối liên hệ với nhau. Các bảng được liên hệ với nhau thông qua phép kết nối của mệnh đề From hoặc thông qua điều kiện của mệnh đề where của câu lệnh select.

Ví dụ: cho mã phòng, tên phòng và tên người lãnh đạo phòng tương ứng.

Trong ví dụ này, tên phòng được lấy từ bảng department, mã phòng có thể lấy từ bảng Department hoặc từ bảng Employee, còn tên nhân viên làm lãnh đạo phòng được lấy từ bảng Employee, hai bảng này được kết nối với nhau thông qua giá trị của thuộc tính Mgr của Department và EmpNo của Employee.

Lưu ý là thuộc tính DeptNo có trong cả hai bảng Department và Employee, do đó khi viết lệnh phải chỉ rõ DeptNo của bảng nào.

```
Select Department.DeptNo, DeptName From  
Department, Employee  
Where Department.Mgr = Employee.EmpNo.
```

Tránh viết nhiều trong câu lệnh SQL hỗ trợ bí danh bằng cách đặt bí danh ngay sau tên bảng nguồn.

```
Câu lệnh trên có thể được viết dưới dạng  
Select D.DeptNo, D.DeptName, E.Name  
From Department D, Employee E  
Where D.Mgr=E.Empno.
```

Kết quả cho ra là bảng:

D.Deptno	D.Deptname	E.Name
10	Kế toán	
20	Marketing	McDonnel
30	Nghiên cứu	Reed
40	Hỗ trợ khách hàng	N Anh
50	Sản xuất	Allen
60	Thành phẩm	

vi. Các câu truy vấn lồng nhau

Nhiều trường hợp chúng ta cần phải tìm kiếm thông tin qua nhiều bước: kết quả của bước trước được sử dụng trong biểu thức của câu truy vấn tiếp theo, rồi kết quả của câu truy vấn tiếp theo nữa v.v...

Ví dụ: cho danh sách những người làm việc cùng phòng với ông Allen

Câu hỏi này có hai bước

a. tìm số hiệu phòng mà ông Allen làm nhân viên (ví dụ phòng tìm được có mã là pp).

b. tìm những người có số hiệu phòng làm việc bằng pp

```
select * from Employeee
```

```
where Deptno= Any(select Deptno
```

```
from employee where Name = 'watson');
```

kết quả là bảng:

EmpNo	Name	Job	Salary	Comm	DeptNo	Sex
102	N Anh	Mngr	3.800		40	F
110	Allen	Slsm	2.700	8.000	40	M

3. Nhóm thao tác tính toán

i. Các hàm tính toán trên nhóm các bản ghi

Các hàm:

Count(*|<tên cột>): đếm số bản ghi có giá trị xác định tại cột được cho bởi <tên cột>.

Sum(<biểu thức>): tính tổng giá trị các biểu thức.

Min(<biểu thức>): tìm giá trị nhỏ nhất.

Max(<biểu thức>): tìm giá trị lớn nhất.

AVG(<biểu thức>): tính giá trị trung bình của biểu thức dựa trên các bản ghi của các nhóm.

Các hàm này thường phải được đi kèm với mệnh đề Group By để thực hiện phân nhóm các bản ghi theo giá trị các cột nào đó trước khi tính toán. Nếu không có mệnh đề Group By thì câu lệnh sẽ coi toàn bộ các bản ghi của bảng là một nhóm.

Ví dụ: cho biết nhân viên có mức lương cao nhất (trung bình) của nhân viên trong công ty.

```
Select Max(salary), AVG(salary)
```

```
From Employee
```

Bảng kết quả:

Max(Salary)	AVG(Salary)
5.300	3.800

ii. *Các hàm tính toán trên bản ghi*

Hầu hết các hệ quản trị CSDL đều cài đặt thư viện hàm xây dựng sẵn trong ngôn ngữ truy vấn dữ liệu nhằm hỗ trợ việc xây dựng các biểu thức tính toán cho từng bộ giá trị (hay bản ghi) của các quan hệ.

- Các hàm toán học
 - o ABS(x): trị tuyệt đối của x.
 - o SQRT(x): căn bậc hai của x
 - o LOG(x): logarit tự nhiên của x
 - o EXP(x): hàm mũ cơ số e của x: e^x
 - o SIGN(x): lấy dấu của số x (trả về 1-: $x < 0$, 0: $x = 0$, +1: $x > 0$)
 - o Round(x,n): làm tròn tới n số lẻ
 - o ... và các hàm lượng giác: Sin, Cos, Tan, Asin, Acos, Atan, ...
- Các hàm xử lý chuỗi ký tự
 - o LEN(str): cho chiều dài dãy ký tự str
 - o Left(str,n) lấy n ký tự phía trái của dãy str.
 - o Right(str,n): lấy n ký tự phía phải của dãy str
 - o MID(str,p,n): lấy n ký tự của dãy str kể từ vị trí p trong dãy.
- Các hàm xử lý ngày tháng và thời gian
 - o Date(): cho ngày tháng năm hiện tại.
 - o Day(dd): cho số thứ tự ngày trong tháng của biểu thức ngày dd
 - o Month(dd): cho số thứ tự tháng trong năm của biểu thức ngày dd.
 - o Year(dd): cho năm của biểu thức ngày dd.
 - o Hour(tt): cho giờ trong ngày (0..23)
 - o Minute(tt): cho số phút của thời gian tt.
 - o Second(tt): cho số giây của biểu thức giờ tt.

4. **Các lệnh khai báo cấu trúc CSDL trong SQL**

Phần này trình bày các lệnh cần thiết để tạo các bảng cùng các ràng buộc toàn vẹn định nghĩa trên các bảng.

i. Cách đặt tên

- Tên gọi gồm tối đa 32 ký tự chữ cái Latinh, chữ số và dấu gạch chân là phải bắt đầu bằng chữ cái Latinh hoặc dấu gạch chân. Tuyệt đối không chứa khoảng trắng hay ký tự chữ cái không phải Latinh như tiếng Việt. Chữ hoa hay chữ thường đều được xem như nhau. Tên bảng phải là duy nhất trong CSDL và tên bảng trung gian và không trùng với bất kỳ từ khóa nào trong ngôn ngữ quản trị CSDL.
- Tên cột của một bảng phải là khác nhau, nhưng chúng có thể giống nhau nếu như nằm trong các bảng khác nhau

ii. Tạo bảng CSDL

Cú pháp:

```
Creat Table<tên bảng>(
<tên cột 1><kiểu dữ liệu 1>(<kích thước 1>),
<tên cột 2><kiểu dữ liệu 2>(<kích thước 2>),
<tên cột n><kiểu dữ liệu n>(<kích thước n>),
);
```

Kiểu dữ liệu có thể là:

- Char(w): kiểu ký tự với kích thước cố định. Chiều dài của giá trị dữ liệu luôn luôn là ký tự. Kích thước tối thiểu là 1 và tối đa là 255 ký tự.
- VarChar(w): Kiểu ký tự với kích thước thay đổi từ 0 đến w ký tự. Giá trị lớn nhất của w là 2000.
- Number(w,s): Kiểu dữ liệu số có kích thước tối đa w ký tự (kể cả dấu chấm thập phân), trong đó có s chữ số sau dấu chấm thể hiện phần số lẻ.
- Date: Kiểu dữ liệu ngày tháng năm.
- Logical: Kiểu dữ liệu logic 1 byte có giá trị hoặc đúng (True), hoặc sai (False).

Ví dụ: khai báo cấu trúc dữ liệu bảng Department trên.

```
Creat Table Department (
```

Deptno Number(2)
Deptname Char (15)
Loc Char (15)
Mgr Number(3)
Exp_Budg Number (7)
Rev_budg Number(7)
);

5. ***Khai báo ràng buộc toàn vẹn***

i. *Ràng buộc toàn vẹn về miền giá trị*

Cú pháp:

[Constraint <tên ràng buộc toàn vẹn>] Check (<điều kiện>)

Ràng buộc toàn vẹn về miền giá trị này có thể định nghĩa trên một cột của bảng, có thể được viết ngay sau tên thuộc tính mà không cần phải viết tên ràng buộc toàn vẹn trong cụm Constraint<tên ràng buộc toàn vẹn>. Nếu định nghĩa trên nhiều cột của bảng thì nên sử dụng cụm này và được viết sau khi đã khai báo xong các thuộc tính của bảng.

<điều kiện> là một biểu thức logic bất kỳ như đã trình bày trong phần câu lệnh truy vấn thông tin, tuy nhiên trong biểu thức không được chứa các câu hỏi con.

Ví dụ: Viết lại câu lệnh khai báo cấu trúc bảng Department với ràng buộc toàn vẹn về miền giá trị cho cột DeptNo:

```
Create Table Department (  
DeptNo Number(2) Primary Ke Check (DeptNo Mod 10=0),  
DeptName Char(15) Not Null,  
Loc Char(15) Not Null Default "Houston",  
Mgr Number(3)  
Exp_Budg Number(7) Not Null,  
Rev_Budg Number(7)  
);
```

Phép toán Mod dùng để lấy phần dư của một phép chia hai số nguyên. Mệnh đề Check ở trên cho phép chỉ nhận những bản ghi có giá trị ở cột DeptNo là bội của 10: 10, 20, 30, .. và 90.

ii. *Ràng buộc toàn vẹn về khóa ngoại hay phụ thuộc tồn tại.*

Cú pháp:

[Constraint<tên ràng buộc toàn vẹn>]
Foreign Key (<các thuộc tính khóa ngoại>)
References<tên bảng> (<các cột khóa chính>)

Ví dụ: Định nghĩa cấu trúc bảng Employee

```
Create Table Employee(  
EmpNo Number(3) Primary Key,  
Name Char(10) Not Null  
Job Char(5) References Jobs(Job),  
Salary Number(5) Not Null,  
Comm Number(5),  
DeptNo Number(2) References Department(DeptNo),  
Sex Char(1) Check(Sex='F' Or Sex ='M')  
);
```

6. Các lệnh quản lý bảng và từ điển dữ liệu

i. Sửa đổi cấu trúc bảng

- Bổ sung thêm cột mới:

Cú pháp:

```
Alter Table<tên bảng>  
Add(<mô tả cột>[<RBTV cột>])  
[mệnh đề Enable|Disable];  
Với <mô tả cột>::=<tên cột> <kiểu d.l>[(<kích thước>)]
```

Ví dụ: Bổ sung thêm cột tên vợ chồng (Spouses_Name) dài 15 ký tự của nhân viên trong bảng Employee:

```
Alter Table Employee  
Add(Spouses_Name char(15));  
- Sửa đổi định nghĩa cột mới
```

Cú pháp:

```
Alter Table<tên bảng>  
Modify(<tên cột><kiểu>(<kích thước>)[Null])  
[mệnh đề Enable|Disable]
```

Lưu ý:

Không thể sửa đổi một cột có chứa giá trị Null từ thuộc tính Null thành Not Null.

Không thể bổ sung một cột mới với thuộc tính Not Null được. Muốn vậy phải thực hiện qua ba bước: (i) thêm cột với thuộc tính Null, (ii) điền đầy đủ các giá trị cho cột, (iii) đổi lại thuộc tính của cột thành Not Null.

Không được phép sửa đổi kích thước của cột cho nhỏ lại, và cũng không được phép thay đổi kiểu dữ liệu của cột, trừ trường hợp cột đó chưa có dữ liệu gì.

Ví dụ: Sửa kích thước cột tên nhân viên (Name) thành 25 ký tự:

```
Alter Table Employee
```

```
Modify(Name char(25));
```

ii. Hủy bỏ bảng

Cú pháp:

```
Drop Table <tên bảng>|<tên View>|<tên Index> [Cascade Constraint];
```

Ngữ nghĩa: Hủy bỏ bảng có tên sau từ khóa Drop Table ra khỏi CSDL. Khi đó tất cả các dữ liệu trong bảng đều bị xóa bỏ. Để hủy bỏ các ràng buộc toàn vẹn khác có liên quan tới bảng, cần phải bổ sung và câu lệnh từ khóa Cascade Constraint.

Ví dụ: hủy bỏ bảng Emplhist khỏi CSDL

```
Drop Table Emplhist;
```

iii. Đổi tên bảng

Cú pháp:

```
Rename<tên bảng cũ>|<tên View cũ> To <tên bảng mới>| <tên View mới>;
```

Ngữ nghĩa: Tên bảng hoặc tên View sẽ được đổi thành tên mới sau từ khóa To.

Ví dụ: đổi tên bảng Department thành Dept.

```
Rename Department To Dept;
```

iv. Xóa bỏ tất cả các dòng của bảng và giải phóng vùng nhớ

Cú pháp:

```
Truncate Table <tên bảng>;
```

Ngữ nghĩa: Khác với Delete From<tên bảng> được sử dụng để xóa bản ghi khỏi bảng nhưng vùng nhớ trên bộ nhớ thứ cấp vẫn không được giải phóng để dùng lại. Lệnh Truncate sau khi đã xóa bỏ các bản ghi khỏi bảng thì vùng bộ nhớ của các bản ghi này sẽ được thu hồi và cho phép các bảng khác sử dụng.

v. Khai báo người sử dụng mới đối với CSDL

Không phải ai cũng có thể tùy tiện truy nhập và CSDL. Người quản trị CSDL sẽ phải thực hiện một lệnh khai báo người sử dụng CSDL và cấp quyền hạn cho họ trên từng bảng của CSDL.

Cú pháp:

Create User<tên người sử dụng> Identified By<mật khẩu truy nhập>;

Ví dụ: Khai báo người sử dụng mới có tên là Minh với mật khẩu là minhkh

Creat User Minh Identify By minhkh;

vi. *Cấp quyền hệ thống trên CSDL*

Grant<quyền hạn>,<quyền hạn>, ... To <người dùng>, <người dùng, ...>;

Trong đó quyền hệ thống gồm có: Create Table, Create View, Create Procedure.

Và lệnh cấp quyền hạn thao tác trên các đối tượng của CSDL với cú pháp:

Grant{<quyền trên đối tượng>[(<cột >, <cột>, ...)|All]

[(<các cột>)]On <đối tượng>

To {<người sử dụng>,<người sử dụng>, ...}| Public

[With Grant Option]);

Ví dụ: cấp quyền truy vấn dữ liệu và cập nhật lại hai file Năm và Sex cho người sử dụng có tên minh trên bảng Employee:

Grant Select, Update(Name, Sex)

On Employee To MINH

vii. *Lấy lại quyền thao tác trên các đối tượng đã trao cho người sử dụng*

Cú pháp:

Revoke{<quyền hạn>,<quyền hạn>, ...|All}

On <đối tượng>

From{<người dùng>,<người dùng>, ...|Public}

[Cascade Constraints];

Mệnh đề Cascade Constraints được sử dụng khi muốn hủy tất cả các ràng buộc toàn vẹn về phụ thuộc tồn tại đã tạo trên đối tượng nhờ quyền hạn References.

Ví dụ: lấy lại quyền truy vấn dữ liệu (select) vừa cập nhật (Update) đã trao cho người sử dụng Minh trên bảng Employee:

Revoke Select, Update

On Employee

From MINH;