

CHEAH SEONG MEN ADDEC004 |

SQL 3 PART 1

Part 1 = Retrieving all columns from a table

1. Customers

SELECT \* FROM customers;

2. Teams

SELECT \* FROM teams;

3. Items

SELECT \* FROM items;

Part 2 = Selecting Specific Columns

1. SELECT ctr-number, first-name, last-name, email, phone-number  
FROM customers;

2. SELECT name, number-of-players  
FROM teams;

3. SELECT name, description, category  
FROM items;

SQL 3 PART 2

Part 1 : Using Arithmetic Operators

1. SELECT first-name, last-name, current-balance,  
current-balance / 12 AS monthly-payment  
FROM customers;

2. SELECT last-name, ctr-number, current-balance,  
current-balance - 5.00 AS adjusted-balance  
FROM customers;

3. When current balance is zero, the 'current balance - 5.00' will become  
-5 which is negative, and it is not allowed in the real world.

### Part 2 = Using Column Aliases

1. `SELECT first-name AS "First Name",  
last-name AS "Last Name",  
current-balance AS "Balance",  
current balance / 12 AS "Monthly Repayments"  
FROM customers;`

### Part 3 = Using Literal Character Strings

1. `SELECT  
'The' || name || 'team has' || number-of-players || 'players and'  
receives a discount of ' || discount || 'percent'  
AS "Team information"  
FROM teams;`
2. It appears that the last team in the data has null value in the discount column. When concatenating the string in the query, the null value is treated as unknown, and it does not display a specific discount in the result.

### SQL 3 PART 3

#### Part 1 = Using the WHERE Clause

1. `SELECT * FROM customers  
WHERE ctr-number = '001986';`
2. `SELECT * first-name AS "First Name",  
last-name AS "Last Name",  
ctr-number AS "Customer Number"  
FROM customers  
WHERE current-balance > 100;`

3. SELECT id AS "Order ID",  
     odr\_date AS "Order Date",  
     odr\_time AS "Order Time"  
 FROM orders  
 WHERE odr\_date < TO\_DATE ('28-May-2019', DD-MM-YYYY);

Part 2 = Range Conditions = BETWEEN Operator

1. SELECT il.id AS "Inventory ID",  
     il.cost AS "Inventory Cost",  
     il.units AS "Number of Units"  
 FROM inventory\_list il  
 WHERE il.cost BETWEEN 3.00 AND 15.00;

Part 3 = Membership Conditions = IN Operator

1. SELECT id AS "Inventory ID",  
     cost AS "Trade Cost",  
     unit AS "Number of Units"  
 FROM inventory\_list  
 WHERE units IN (50, 100, 150, 200);

Part 4 = Membership Conditions = NOT IN Operator

1. SELECT id AS "Inventory ID",  
     cost AS "Trade Cost",  
     units AS "Number of Units"  
 FROM inventory\_list  
 WHERE units NOT IN (50, 100, 150, 200);

Part 5 = Pattern Matching = LIKE Operator

1. SELECT item\_number AS "Item Number",  
     name AS "Item Name"  
 FROM items  
 WHERE name like 'g%';

Part 6 = Pattern Matching = Combining Wildcard Characters with the LIKE Operator

1. SELECT item-number AS "Item Number",  
name AS "Item Name",  
FROM items  
WHERE name LIKE '%o%';

SQL 3 PART 4

Part 1 = Using the NULL Condition

1. SELECT  
'The' || name || ' team has ' || number\_of\_players ||  
' players and does not receive a discount.' AS "Team Information"  
FROM teams  
WHERE discount IS NULL;

2. SELECT

'The' || name || ' team has ' || number\_of\_players || ' players and  
receives a discount of ' || discount || ' percent.' AS "Team Information"  
FROM teams  
WHERE discount IS NOT NULL;

Part 2 = Logical Operators : AND

1. SELECT ctr-number AS "Customer Number",  
address-line-1 AS "Street Address",  
zip-code AS "Postal Code"  
FROM customers\_addresses  
WHERE city = 'Liverpool' AND address-line-2 = 'Starford';

Part 3 = Logical Operator = OR

1. SELECT ctr-number AS "Customer Number",  
address-line-2 AS "Street Address",  
zip-code AS "Postal Code"  
FROM customers\_addresses  
WHERE city = 'Liverpool' OR address-line-2 = 'Starford';

Part 4 = Logical Operators = NOT Equal to

1. SELECT ctr-number AS "Customer Number",  
 address-line-1 AS "Street Address",  
 zip-code AS "Postal Code"  
 FROM customers\_address  
 WHERE city NOT LIKE 'Liverpool';

SQL 3 PART 5

Use the ORDER BY Clause to Sort SQL Results

1. SELECT name AS "Team Name",  
 number\_of\_players AS "Number of Players"  
 FROM teams

ORDER BY "Team Name" ASC;

2. SELECT name AS "Team Name",  
 number\_of\_players AS "Number of Players"  
 FROM teams

ORDER BY "Number of Players" DESC;

3. SELECT name AS "Team Name",  
 number\_of\_players AS "Players"

FROM teams

ORDER BY "Team Name" DESC;

SQL 3 PART 6

Part 1 = TOP-N-ANALYSIS

SELECT first-name || '' || last-name AS "Customer Name"

FROM (

SELECT first-name, last-name, ROWNUM AS rnum

FROM customers

ORDER BY ctr-number

)

WHERE rnum <= 3;

Part 2 = Using a Substitution Variable

SELECT first\_name AS "First Name",

last\_name AS "Last Name"

FROM sales\_representatives

WHERE comission\_rate = :input\_comission\_rate

ORDER BY last\_name;

## SQL 3 PART 1

### Part 1: Retrieving all columns from a table

#### 1. Customers

```
SELECT * FROM customers;
```

CTR_NUMBER	EMAIL	FIRST_NAME	LAST_NAME	PHONE_NUMBER	CURRENT_BALANCE	SRE_ID	TEM_ID	LOYALTY_CARD_NUMBER
1 c00001	bob.thornberry@heatmail.com	Robert	Thornberry	01234567898	150	sr01	t001	(null)
2 c00012	Jjones@freemail.com	Jennifer	Jones	01505214598	0	(null)	(null)	lc1015
3 c00101	unknown@here.com	John	Doe	03216547808	987.5	sr01	t002	(null)
4 c00103	MurciaA@globaltech.com	Andrew	Murcia	07715246890	85	(null)	(null)	lc2341
5 c01986	margal87@delphiview.com	Maria	Galant	01442736589	125.65	sr03	t003	(null)
6 c02001	brianrog@hootech.com	Brian	Roger	01654564898	50	(null)	(null)	lc4587

#### 2. Teams

```
SELECT * FROM teams;
```

ID	NAME	NUMBER_OF_PLAYERS	DISCOUNT
1 t001	Rockets	25	10
2 t002	Celtics	42	20
3 t003	Rovers	8	(null)
4 t004	Jets	10	5

#### 3. Items

```
SELECT * FROM items;
```

ITM_NUMBER	NAME	DESCRIPTION	CATEGORY	COLOR	Size	ILT_ID
1 im01101044	gloves	catcher mitt	clothing	brown	m	i1010230124
2 im01101045	under shirt	top worn under the game top	clothing	white	s	i1010230125
3 im01101046	socks	team socks with emblem	clothing	range	l	i1010230126
4 im01101047	game top	team shirt with emblem	clothing	range	m	i1010230127
5 im01101048	premium bat	high quality baseball bat	equipment	(null)	(null)	i1010230128

### Part 2: Selecting Specific Columns

1. SELECT ctr\_number, first\_name, last\_name, email, phone\_number  
FROM customers;

CTR_NUMBER	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER
1 c00001	Robert	Thornberry	bob.thornberry@heatmail.com	01234567898
2 c00012	Jennifer	Jones	Jjones@freemail.com	01505214598
3 c00101	John	Doe	unknown@here.com	03216547808
4 c00103	Andrew	Murcia	MurciaA@globaltech.com	07715246890
5 c01986	Maria	Galant	margal87@delphiview.com	01442736589
6 c02001	Brian	Roger	brianrog@hootech.com	01654564898

2. SELECT name, number\_of\_players  
FROM teams;

	NAME	NUMBER_OF_PLAYERS
1	Rockets	25
2	Celtics	42
3	Rovers	8
4	Jets	10

3. SELECT name, description, category  
FROM items;

	NAME	DESCRIPTION	CATEGORY
1	gloves	catcher mitt	clothing
2	under shirt	top worn under the game top	clothing
3	socks	team socks with emblem	clothing
4	game top	team shirt with emblem	clothing
5	premium bat	high quality baseball bat	equipment

## SQL 3 PART 2

## Part 1: Using Arithmetic Operators

1.

## SELECT

```
first_name,  
last_name,  
current_balance,  
current_balance / 12 AS monthly_payment
```

FROM customers;

2.

## SELECT

```
first_name,  
last_name,  
ctr_number,  
current_balance,  
current balance - 5.00 AS adjusted balance
```

FROM customers;

	FIRST_NAME	LAST_NAME	CTR_NUMBER	CURRENT_BALANCE	ADJUSTED_BALANCE
1	Robert	Thornberry	c00001	150	145
2	Jennifer	Jones	c00012	0	-5
3	John	Doe	c00101	987.5	982.5
4	Andrew	Murcia	c00103	85	80
5	Maria	Galant	c01986	125.65	120.65
6	Brian	Roger	c02001	50	45

3.

## Part 2 : Using Column Aliases

1.

SELECT

```
first_name AS "First Name",
last_name AS "Last Name",
current_balance AS "Balance",
current_balance / 12 AS "Monthly Repayments"
```

FROM customers;

	First Name	Last Name	Balance	Monthly Repayments
1	Robert	Thornberry	150	12.5
2	Jennifer	Jones	0	0
3	John	Doe	987.5	82.2916667
4	Andrew	Murcia	85	7.08333
5	Maria	Galant	125.65	10.4708333
6	Brian	Roger	50	4.16667

## Part 3: Using Literal Character Strings

1.

SELECT

```
'The ' || name || ' team has ' || number_of_players || ' players and receives a discount of
' || discount || ' percent.' AS "Team Information"
```

FROM teams;

	Team Information
1	The Rockets team has 25 players and receives a discount of 10 percent.
2	The Celtics team has 42 players and receives a discount of 20 percent.
3	The Rovers team has 8 players and receives a discount of percent.
4	The Jets team has 10 players and receives a discount of 5 percent.

2. It appears that the last team in the data has a null value in the discount column. When concatenating the string in the query, the null value is treated as unknown, and it does not display a specific discount in the result.

## SQL 3 PART 3

### Part 1: Using the WHERE Clause

1.

```
SELECT *  
FROM customers  
WHERE ctr_number = 'c01986';
```

	CTR_NUMBER	EMAIL	FIRST_NAME	LAST_NAME	PHONE_NUMBER	CURRENT_BALANCE	SRE_ID	TEM_ID	LOYALTY_CARD_NUMBER
1	c01986	margal87@delphiview.com	Maria	Galant	01442736589	125.65	sr03	t003	(null)

2.

```
SELECT  
    first_name AS "First Name",  
    last_name AS "Last Name",  
    ctr_number AS "Customer Number"  
  
FROM customers  
  
WHERE current_balance > 100;
```

	First Name	Last Name	Customer Number
1	Robert	Thornberry	c00001
2	John	Doe	c00101
3	Maria	Galant	c01986

3.

```
SELECT  
    id AS "Order ID",  
    odr_date AS "Order Date",  
    odr_time AS "Order Time"  
  
FROM orders  
  
WHERE odr_date < TO_DATE('28-May-2019', 'DD-MM-YYYY');
```

	Order ID	Order Date	Order Time
1	or0101250	17-APR-17	17-APR-17
2	or0101350	24-MAY-17	24-MAY-17
3	or0101425	28-MAY-17	28-MAY-17
4	or0101681	02-JUN-17	02-JUN-17
5	or0101750	18-JUN-17	18-JUN-17

## Part 2: Range Conditions: BETWEEN Operator

```
SELECT  
    il.id AS "Inventory ID",  
    il.cost AS "Trade Cost",  
    il.units AS "Number of Units"  
FROM inventory_list il  
WHERE il.cost BETWEEN 3.00 AND 15.00;
```

	Inventory ID	Trade Cost	Number of Units
1	i1010230125	7.99	250
2	i1010230126	5.24	87

## Part 3: Membership Conditions: IN Operator

```
SELECT  
    id AS "Inventory ID",  
    cost AS "Trade Cost",  
    units AS "Number of Units"  
FROM inventory_list  
WHERE units IN (50, 100, 150, 200);
```

	Inventory ID	Trade Cost	Number of Units
1	i1010230124	2.5	100

## Part 4: Membership Conditions: NOT IN Operator

```
SELECT  
    id AS "Inventory ID",  
    cost AS "Trade Cost",  
    units AS "Number of Units"  
FROM inventory_list  
WHERE units NOT IN (50, 100, 150, 200);
```

	Inventory ID	Trade Cost	Number of Units
1	il010230125	7.99	250
2	il010230126	5.24	87
3	il010230127	18.95	65
4	il010230128	97.46	8

#### Part 5: Pattern Matching: LIKE Operator

1.

SELECT

```
itm_number AS "Item Number",
name AS "Item Name"
FROM items
WHERE name LIKE 'g%';
```

	Item Number	Item Name
1	im01101044	gloves
2	im01101047	game top

#### Part 6 : Pattern Matching: Combining Wildcard Characters with the LIKE Operator

1.

SELECT

```
itm_number AS "Item Number",
name AS "Item Name"
FROM items
WHERE name LIKE '%o%';
```

	Item Number	Item Name
1	im01101044	gloves
2	im01101046	socks
3	im01101047	game top

## SQL 3 PART 4

### Part 1: Using the NULL Conditions

SELECT

```
'The ' || name || ' team has ' || number_of_players || ' players and does not receive a discount.' AS "Team Information"
```

FROM teams

WHERE discount IS NULL;

Team Information
1 The Rovers team has 8 players and does not receive a discount.

2.

SELECT

```
'The ' || name || ' team has ' || number_of_players || ' players and receives a discount of ' || discount || ' percent.' AS "Team Information"
```

FROM teams

WHERE discount IS NOT NULL;

Team Information
1 The Rockets team has 25 players and receives a discount of 10 percent.
2 The Celtics team has 42 players and receives a discount of 20 percent.
3 The Jets team has 10 players and receives a discount of 5 percent.

### Part 2: Logical Operators: AND

1.

SELECT

```
ctr_number AS "Customer Number",  
address_line_1 AS "Street Address",  
zip_code AS "Postal Code"
```

FROM customers\_addresses

WHERE city = 'Liverpool' AND address\_line\_2 = 'Starford';

Customer Number	Street Address	Postal Code
1 c00001	17 Gartsquare Road	LP89JHK

### Part 3: Logical Operators: OR

1.

SELECT

```
ctr_number AS "Customer Number",
address_line_1 AS "Street Address",
zip_code AS "Postal Code"

FROM customers_addresses

WHERE city = 'Liverpool' OR address_line_2 = 'Starford';
```

	Customer Number	Street Address	Postal Code
1	c00001	17 Gartsquare Road	LP89JHK
2	c00001	63 Acacia Drive	LP83JHR

### Part 4: Logical Operators: NOT Equal To

1.

SELECT

```
ctr_number AS "Customer Number",
address_line_1 AS "Street Address",
zip_code AS "Postal Code"

FROM customers_addresses

WHERE city NOT LIKE 'Liverpool';
```

	Customer Number	Street Address	Postal Code
1	c00101	54 Ropehill Crescent	ST45AGV
2	c01986	36 Watercress Lane	JP23YTH

## SQL 3 PART 5

### Use the ORDER BY Clause to Sort SQL Results

1.

SELECT

```
name AS "Team Name",
number_of_players AS "Number of Players"
FROM teams
ORDER BY "Team Name" ASC;
```

	Team Name	Number of Players
1	Celtics	42
2	Jets	10
3	Rockets	25
4	Rovers	8

2.

SELECT

```
name AS "Team Name",
number_of_players AS "Number of Players"
FROM teams
ORDER BY "Number of Players" DESC;
```

	Team Name	Number of Players
1	Celtics	42
2	Rockets	25
3	Jets	10
4	Rovers	8

3.

SELECT

```
name AS "Team Name",
number_of_players AS "Players"
FROM teams
ORDER BY "Team Name" DESC;
```

	Team Name	Players
1	Rovers	8
2	Rockets	25
3	Jets	10
4	Celtics	42

## SQL 3 PART 6

### Part 1 : TOP-N-ANALYSIS (S6L8 Objective 3)

```
SELECT
    first_name || ' ' || last_name AS "Customer Name"
FROM (
    SELECT
        first_name,
        last_name,
        ROWNUM AS rnum
    FROM customers
    ORDER BY ctr_number
)
WHERE rnum <= 3;
```

Customer Name	
1	Robert Thornberry
2	Jennifer Jones
3	John Doe

### Part 2 : Using a Substitution Variable (S6L8 Objective 4)

```
SELECT
    first_name AS "First Name",
    last_name AS "Last Name"
FROM sales_representatives
WHERE commission_rate = :input_commission_rate
ORDER BY last_name;
```

For the input = 5

	First Name	Last Name
1	Barry	Speed
2	Victoria	Wright