Question 1

Display the last name concatenated with the job ID, separated by a comma and space, and name the column Employee and Title.

Answer

SQL Worksheet

```
1  SELECT LAST_NAME || ', ' || JOB_ID AS "Employee and Title" FROM hr.employees;
2
```

Output

Employee and Title
Abel, SA_REP
Ande, SA_REP
Atkinson, ST_CLERK
Austin, IT_PROG
Baer, PR_REP
Baida, PU_CLERK
Banda, SA_REP
Bates, SA_REP
Bell, SH_CLERK
Bernstein, SA_REP
Bissot, ST_CLERK
Bloom, SA_REP
Bull, SH_CLERK
Cabrio, SH_CLERK
Cambrault, SA_MAN
Cambrault, SA_REP

Question 2

Create a query to display all the data from the EMPLOYEES table. Separate each column by a comma. Name the column THE_OUTPUT.

Answer 2

SQL Worksheet

```
1
2  SELECT EMPLOYEE_ID || ', ' || FIRST_NAME || ', ' || LAST_NAME || ', ' || EMAIL || ', ' || PHONE_NUMBER || ', ' || HIRE_DATE || ', '
3  || JOB_ID || ', ' || SALARY || ', ' || COMMISSION_PCT
4  || ', ' || MANAGER_ID || ', ' || DEPARTMENT_ID AS THE_OUTPUT FROM hr.employees;
5
```

Output

THE_OUTPUT
100, Steven, King, SKING, 515.123.4567, 17-JUN-03, AD_PRES, 24000, , , 90
101, Neena, Kochhar, NKOCHHAR, 515.123.4568, 21-SEP-05, AD_VP, 17000, , 100, 90
102, Lex, De Haan, LDEHAAN, 515.123.4569, 13-JAN-01, AD_VP, 17000, , 100, 90
103, Alexander, Hunold, AHUNOLD, 590.423.4567, 03-JAN-06, IT_PROG, 9000, , 102, 60
104, Bruce, Ernst, BERNST, 590.423.4568, 21-MAY-07, IT_PROG, 6000, , 103, 60
105, David, Austin, DAUSTIN, 590.423.4569, 25-JUN-05, IT_PROG, 4800, , 103, 60
106, Valli, Pataballa, VPATABAL, 590.423.4560, 05-FEB-06, IT_PROG, 4800, , 103, 60
107, Diana, Lorentz, DLORENTZ, 590.423.5567, 07-FEB-07, IT_PROG, 4200, , 103, 60
108, Nancy, Greenberg, NGREENBE, 515.124.4569, 17-AUG-02, FI_MGR, 12008, , 101, 100
109, Daniel, Faviet, DFAVIET, 515.124.4169, 16-AUG-02, FI_ACCOUNT, 9000, , 108, 100
110, John, Chen, JCHEN, 515.124.4269, 28-SEP-05, FI_ACCOUNT, 8200, , 108, 100
111, Ismael, Sciarra, ISCIARRA, 515.124.4369, 30-SEP-05, FI_ACCOUNT, 7700, , 108, 100
112, Jose Manuel, Urman, JMURMAN, 515.124.4469, 07-MAR-06, FI_ACCOUNT, 7800, , 108, 100
113, Luis, Popp, LPOPP, 515.124.4567, 07-DEC-07, FI_ACCOUNT, 6900, , 108, 100
114, Den, Raphaely, DRAPHEAL, 515.127.4561, 07-DEC-02, PU_MAN, 11000, , 100, 30
115, Alexander, Khoo, AKHOO, 515.127.4562, 18-MAY-03, PU_CLERK, 3100, , 114, 30

Question 3

Show the structure of the DEPARTMENTS table. Select all data from the table.

Answer 3

SQL Worksheet

```
1  describe hr.departments;
2  select * from hr.departments;
```

Output

TABLE DEPARTMENTS

Column	Null?	Type
DEPARTMENT_ID	NOT NULL	NUMBER(4,0)
DEPARTMENT_NAME	NOT NULL	VARCHAR2(30)
MANAGER_ID	-	NUMBER(6,0)
LOCATION_ID	-	NUMBER(4,0)

Download CSV

4 rows selected.

DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
10	Administration	200	1700
20	Marketing	201	1800
30	Purchasing	114	1700
40	Human Resources	203	2400
50	Shipping	121	1500
60	IT	103	1400
70	Public Relations	204	2700
80	Sales	145	2500
90	Executive	100	1700
100	Finance	108	1700
110	Accounting	205	1700
120	Treasury	-	1700
130	Corporate Tax	-	1700
140	Control And Credit	-	1700
150	Shareholder Services	-	1700

Question 4

Show the structure of the EMPLOYEES table. Create a query to display the last name, job code, hire date, and employee number for each employee, with employee number appearing first.

Answer 4

SQL Worksheet

```
1  select last_name, job_id, hire_date, phone_number from hr.employees;
```

Output

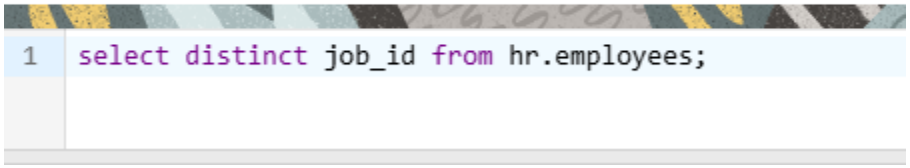
LAST_NAME	JOB_ID	HIRE_DATE	PHONE_NUMBER
King	AD_PRES	17-JUN-03	515.123.4567
Kochhar	AD_VP	21-SEP-05	515.123.4568
De Haan	AD_VP	13-JAN-01	515.123.4569
Hunold	IT_PROG	03-JAN-06	590.423.4567
Ernst	IT_PROG	21-MAY-07	590.423.4568
Austin	IT_PROG	25-JUN-05	590.423.4569
Pataballa	IT_PROG	05-FEB-06	590.423.4560
Lorentz	IT_PROG	07-FEB-07	590.423.5567
Greenberg	FI_MGR	17-AUG-02	515.124.4569
Faviet	FI_ACCOUNT	16-AUG-02	515.124.4169
Chen	FI_ACCOUNT	28-SEP-05	515.124.4269
Sciarra	FI_ACCOUNT	30-SEP-05	515.124.4369
Urman	FI_ACCOUNT	07-MAR-06	515.124.4469
Popp	FI_ACCOUNT	07-DEC-07	515.124.4567
Raphaely	PU_MAN	07-DEC-02	515.127.4561
Khoo	PU_CLERK	18-MAY-03	515.127.4562

Question 5

Create a query to display unique job codes from the EMPLOYEES table.

Answer 5

SQL Worksheet



Output

JOB_ID
AC_ACCOUNT
AC_MGR
AD_ASST
AD_PRES
AD_VP
FI_ACCOUNT
FI_MGR
HR_REP
IT_PROG
MK_MAN
MK_REP
PR_REP
PU_CLERK
PU_MAN

Question 6

There are four coding errors in this statement. Can you identify them?

SELECT employee_id, last_name sal x 12 ANNUAL SALARY FROM employees;

Answer 6

sal x 12 ANNUAL SALARY is not valid, rather it should be sal * 12 AS ANNUAL_SALARY, using the multiplication operator and AS statement to change the sal*12 column to be named as Annual salary. There should be a comma after last name and sal*12 columns so that they both can be displayed properly.Moreover, ANNUAL SALARY contains a space, which is not valid unless wrapped in double quotes or square brackets. We can fix it by using AS "ANNUAL SALARY" or change it to a valid name like ANNUAL_SALARY.

Question 7

1757. Recyclable and Low Fat Products

```
+-------------+---------+
| Column Name | Type    |
+-------------+---------+
| product_id  | int     |
| low_fats    | enum    |
| recyclable  | enum    |
+-------------+---------+
product_id is the primary key (column with unique values) for this table.
low_fats is an ENUM (category) of type ('Y', 'N') where 'Y' means this product is low fat and 'N' means
it is not.
recyclable is an ENUM (category) of types ('Y', 'N') where 'Y' means this product is recyclable and 'N'
means it is not.
```

Code

MySQL Auto

```mysql
# Write your MySQL query statement below
select product_id
from products
where low_fats = 'Y' and recyclable = 'Y'
```

Saved

Testcase | Test Result

Accepted Runtime: 187 ms

• Case 1

Input

Products =

product_id	low_fats	recyclable
0	Y	N
1	Y	Y
2	N	Y
3	Y	Y
4	N	N

Output

product_id
1
3

Expected

product_id
1
3

584. Find Customer Referee

Easy | Topics | Companies | Hint

SQL Schema > Pandas Schema >

Table: `Customer`

```
+-------------+---------+
| Column Name | Type    |
+-------------+---------+
| id          | int     |
| name        | varchar |
| referee_id  | int     |
+-------------+---------+
In SQL, id is the primary key column for this table.
Each row of this table indicates the id of a customer, their name, and the id of the customer who
referred them.
```

Find the names of the customer that are **not referred by** the customer with `id = 2`.

Return the result table in **any order**.

The result format is in the following example.

Code

MySQL Auto

```mysql
# Write your MySQL query statement below
SELECT name
FROM Customer
WHERE referee_id IS NULL OR referee_id != 2;
```

Saved Ln 4, Co

Testcase | Test Result

Accepted Runtime: 197 ms

• Case 1

Input

Customer =

id	name	referee_id
1	Will	null
2	Jane	null
3	Alex	2
4	Bill	null
5	Zack	1
6	Mark	2

Output

name
Will
Jane
Bill
Zack

Description | Editorial | Solutions | Submissions

595. Big Countries

Easy | Topics | Companies

SQL Schema > Pandas Schema >

Table: `World`

```
+-------------+---------+
| Column Name | Type    |
+-------------+---------+
| name        | varchar |
| continent   | varchar |
| area        | int     |
| population  | int     |
| gdp         | bigint  |
+-------------+---------+
name is the primary key (column with unique values) for this table.
Each row of this table gives information about the name of a country, the continent to which it belongs,
its area, the population, and its GDP value.
```

A country is **big** if:

- it has an area of at least three million (i.e., `3000000 km²`), or
- it has a population of at least twenty-five million (i.e., `25000000`).

Write a solution to find the name, population, and area of the **big countries**.

Return the result table in **any order**.

The result format is in the following example.

Code

MySQL Auto

```mysql
# Write your MySQL query statement below
select name, population, area
from world
where area>=3000000 || population >=25000000;
```

Saved Ln 4

Testcase | Test Result

Accepted Runtime: 212 ms

Case 1

Input

World =

name	continent	area	population	gdp
Afghanistan	Asia	652230	25500100	20343000000
Albania	Europe	28748	2831741	12960000000
Algeria	Africa	2381741	37100000	188681000000
Andorra	Europe	468	78115	3712000000
Angola	Africa	1246700	20609294	100990000000

Output

name	population	area
Afghanistan	25500100	652230
Algeria	37100000	2381741

Expected

name	population	area
Afghanistan	25500100	652230
Algeria	37100000	2381741

Description | Editorial | Solutions | Submissions

1148. Article Views I

Easy | Topics | Companies

SQL Schema > Pandas Schema >

Table: `Views`

Column Name	Type
article_id	int
author_id	int
viewer_id	int
view_date	date

There is no primary key (column with unique values) for this table, the table may have duplicate rows. Each row of this table indicates that some viewer viewed an article (written by some author) on some date.
Note that equal author_id and viewer_id indicate the same person.

Write a solution to find all the authors that viewed at least one of their own articles.

Return the result table sorted by `id` in ascending order.

The result format is in the following example.

Code

MySQL Auto

```mysql
# Write your MySQL query statement below
SELECT DISTINCT author_id AS id
FROM Views
WHERE author_id = viewer_id
ORDER BY 1;
```

Saved Ln 5, Col

Testcase | Test Result

Accepted Runtime: 154 ms

• Case 1

Input

Views =

article_id	author_id	viewer_id	view_date
1	3	5	2019-08-01
1	3	6	2019-08-02
2	7	7	2019-08-01
2	7	6	2019-08-02
4	7	1	2019-07-22
3	4	4	2019-07-21

View more

Output

id
4
7

Description | Editorial | Solutions | Submissions

1683. Invalid Tweets

Easy | Topics | Companies

SQL Schema > Pandas Schema >

Table: `Tweets`

```
+----------------+---------+
| Column Name    | Type    |
+----------------+---------+
| tweet_id       | int     |
| content        | varchar |
+----------------+---------+
tweet_id is the primary key (column with unique values) for this table.
content consists of characters on an American Keyboard, and no other special characters.
This table contains all the tweets in a social media app.
```

Write a solution to find the IDs of the invalid tweets. The tweet is invalid if the number of characters used in the content of the tweet is **strictly greater** than `15`.

Return the result table in **any order**.

The result format is in the following example.

Code

MySQL Auto

```mysql
# Write your MySQL query statement below
SELECT tweet_id
FROM Tweets
WHERE CHAR_LENGTH(content) > 15;
```

Saved Ln 4, Col 33

Testcase | Test Result

Accepted Runtime: 199 ms

• Case 1

Input

Tweets =

tweet_id	content
1	Let us Code
2	More than fifteen chars are here!

Output

tweet_id
2

Expected

tweet_id
2

Contribute a testcase