

## Question 1

a) Create three tables based on the below schema and assign a primary key of your choice for each table. (3)

The screenshot shows the Oracle Online SQL Editor interface. On the left, there are four tabs: Assignments, Customers, Employees, and Orders. The Assignments tab is selected, displaying its schema:

```
CREATE TABLE Employees (EmployeeID INTEGER, Name VARCHAR(10), Department VARCHAR(10), Salary INTEGER, HireDate DATE FORMAT 'YYYY-MM-DD');
```

The Customers tab displays:

```
CREATE TABLE Projects (ProjectID INTEGER, Name VARCHAR(10), Department VARCHAR(10), StartDate DATE FORMAT 'YYYY-MM-DD', EndDate DATE FORMAT 'YYYY-MM-DD');
```

The Employees tab displays:

```
CREATE TABLE Customers (customer_id INT, first_name VARCHAR(100), last_name VARCHAR(100), age INT, country VARCHAR(100));
```

The Orders tab displays:

```
CREATE TABLE Orders (order_id INT, item VARCHAR(100));
```

In the center, the Input field contains the following SQL code:

```
CREATE TABLE Employees (EmployeeID INTEGER, Name VARCHAR(10), Department VARCHAR(10), Salary INTEGER, HireDate DATE FORMAT 'YYYY-MM-DD');
INSERT INTO Employees (EmployeeID, Name, Department, Salary, HireDate)
VALUES (1, 'John Doe', 'Engineering', 60000, '2022-01-05'), (2, 'Alice Smith', 'Marketing', 55000, '2021-03-12'), (3, 'Bob Johnson', 'Engineering', 65000, '2023-05-20'), (4, 'Sarah Williams', 'HR', 50000, '2022-09-10'), (5, 'Michael Brown', 'Finance', 70000, '2021-11-15');

CREATE TABLE Projects (ProjectID INTEGER, Name VARCHAR(10), Department VARCHAR(10), StartDate DATE FORMAT 'YYYY-MM-DD', EndDate DATE FORMAT 'YYYY-MM-DD');
INSERT INTO Projects (ProjectID, Name, Department, StartDate, EndDate)
VALUES (1, 'Website Redesign Engineering', 'Engineering', '2022-02-10', '2022-08-15'), (2, 'Product Launch Marketing', 'Marketing', '2021-05-20', '2021-10-30'), (3, 'Infrastructure Upgrade Engineering', 'Engineering', '2023-01-15', '2023-07-30'), (4, 'Employee Training Program', 'HR', '2022-10-01', '2023-03-31'), (5, 'Budget Planning Finance', 'Finance', '2022-12-01', '2023-06-30');
```

The Available Tables section shows three tables: Assignments, Customers, and Employees. The Employees table has the following data:

AssignmentID	EmployeeID	ProjectID	HoursWorked
1	1	1	120.5
2	2	2	90
3	3	1	150.25
4	4	4	80.75
5	5	5	110

The Customers table has the following data:

customer_id	first_name	last_name	age	country
1	John	Doe	31	USA
2	Robert	Luna	22	USA
3	David	Robinson	22	UK
4	John	Reinhardt	25	UK
5	Betty	Doe	28	UAE

The Employees table has the following data:

EmployeeID	Name	Department	Salary	HireDate
1	John	Engineering	60000	2022-01-05

At the bottom, the status bar shows: 2°C Cloudy, Search, various icons, ENG US, 5:47 PM, 2024-02-11.

Online SQL Editor <https://www.programiz.com/sql/online-compiler/>

**Programiz** Online SQL Editor

**Input**

```

2023-01-20),(4, Sarah Williams , HR , 30000, 2022-09-10),(5, Michael Brown,'Finance', 70000, '2021-11-15');

CREATE TABLE Projects (ProjectID INTEGER, Name VARCHAR(10), Department VARCHAR(10), StartDate DATE FORMAT 'YYYY-MM-DD', EndDate DATE FORMAT 'YYYY-MM-DD');
INSERT INTO Projects (ProjectID, Name, Department, StartDate, EndDate)
VALUES (1, 'Website Redesign Engineering','Engineering', '2022-02-10', '2022-08-15'),(2,'Product Launch Marketing','Marketing', '2021-05-20', '2021-10-30'),(3,'Infrastructure Upgrade Engineering','Engineering', '2023-01-15', '2023-07-30'),(4,'Employee Training Program','HR', '2022-10-01', '2023-03-31'),(5, 'Budget Planning' , 'Finance', '2022-12-05', '2023-06-30');

CREATE TABLE Assignments (AssignmentID INTEGER, EmployeeID INTEGER, ProjectID INTEGER, Hoursworked INTEGER);
INSERT INTO Assignments (AssignmentID , EmployeeID , ProjectID, Hoursworked)
VALUES (1,1,120.5),(2,2,2,90.0),(3,3,1,150.25),(4,4,4,80.75),(5,5,5,110.0);

```

**Available Tables**

**Employees**

EmployeeID	Name	Department	Salary	HireDate
1	John Doe	Engineering	60000	2022-01-05
2	Alice Smith	Marketing	55000	2021-03-12
3	Bob Johnson	Engineering	65000	2023-05-20
4	Sarah Williams	HR	50000	2003
5	Michael Brown	Finance	70000	2021-11-15

**Orders**

order_id	item	amount	customer_id
1	Keyboard	400	4
2	Mouse	300	4
3	Monitor	12000	3
4	Keyboard	400	1

**Output**

SQL query successfully executed. However, the result set is empty.

Online SQL Editor <https://www.programiz.com/sql/online-compiler/>

**Programiz** Online SQL Editor

**Input**

```

2023-01-20),(4, Sarah Williams , HR , 30000, 2022-09-10),(5, Michael Brown,'Finance', 70000, '2021-11-15');

CREATE TABLE Projects (ProjectID INTEGER, Name VARCHAR(10), Department VARCHAR(10), StartDate DATE FORMAT 'YYYY-MM-DD', EndDate DATE FORMAT 'YYYY-MM-DD');
INSERT INTO Projects (ProjectID, Name, Department, StartDate, EndDate)
VALUES (1, 'Website Redesign Engineering','Engineering', '2022-02-10', '2022-08-15'),(2,'Product Launch Marketing','Marketing', '2021-05-20', '2021-10-30'),(3,'Infrastructure Upgrade Engineering','Engineering', '2023-01-15', '2023-07-30'),(4,'Employee Training Program','HR', '2022-10-01', '2023-03-31'),(5, 'Budget Planning' , 'Finance', '2022-12-05', '2023-06-30');

CREATE TABLE Assignments (AssignmentID INTEGER, EmployeeID INTEGER, ProjectID INTEGER, Hoursworked INTEGER);
INSERT INTO Assignments (AssignmentID , EmployeeID , ProjectID, Hoursworked)
VALUES (1,1,120.5),(2,2,2,90.0),(3,3,1,150.25),(4,4,4,80.75),(5,5,5,110.0);

```

**Available Tables**

**Projects**

ProjectID	Name	Department	StartDate	EndDate
1	Website Redesign Engineering	Engineering	2022-02-10	2022-08-15
2	Product Launch Marketing	Marketing	2021-05-20	2021-10-30
3	Infrastructure Upgrade Engineering	Engineering	2023-01-15	2023-07-30
4	Employee Training Program	HR	2022-10-01	2023-03-31
5	Budget Planning	Finance	2022-12-05	2023-06-30

**Shipments**

shipping_id	status	customer
-------------	--------	----------

**Output**

SQL query successfully executed. However, the result set is empty.

b) Calculate the total hours worked by each employee. Retrieve the name and total hours worked. (3)

The screenshot shows the Oracle Online SQL Editor interface. On the left, there are four tabs: 'Assignments [-]', 'Customers [-]', 'Employees [-]', and 'Orders [-]'. The 'Input' tab is selected, displaying the following SQL code:

```
--CREATE TABLE Assignments (AssignmentID INTEGER, EmployeeID INTEGER, ProjectID INTEGER, HoursWorked INTEGER);
--INSERT INTO Projects (ProjectID, Name, Department, StartDate, EndDate)
--VALUES (1, 'Website Redesign Engineering', 'Engineering', '2022-02-10', '2022-08-15'),(2, 'Product Launch Marketing', 'Marketing', '2021-05-20', '2021-10-30'),(3,'Infrastructure Upgrade Engineering', 'Engineering', '2023-01-15', '2023-07-30'),(4,'Employee Training Program', 'HR', '2022-10-01', '2023-03-31'),(5, 'Budget Planning', 'Finance', '2022-12-05', '2023-06-30');

--CREATE TABLE Assignments (AssignmentID INTEGER, EmployeeID INTEGER, ProjectID INTEGER, HoursWorked INTEGER);
--INSERT INTO Assignments (AssignmentID , EmployeeID , ProjectID, HoursWorked)
--VALUES (1,1,120.5),(2,2,90.0),(3,3,1,150.25),(4,4,4,80.75),(5,5,5,110.0);

SELECT Employees.Name, Assignments.HoursWorked
FROM Employees
JOIN Assignments ON Employees.EmployeeID = Assignments.EmployeeID;
```

The 'Available Tables' section lists the 'Assignments', 'Customers', 'Employees', and 'Orders' tables. The 'Output' section displays the results of the query:

Name	HoursWorked
John Doe	120.5
Alice Smith	90
Bob Johnson	150.25
Sarah Williams	80.75
Michael Brown	110

C) Calculate the average salary by department. (3)

Online SQL Editor

https://www.programiz.com/sql/online-compiler/

**Programiz** Online SQL Editor

How do top CIOs build AI-enabled enterprises? Get the ebook

Interactive SQL Course

**Input**

```
--VALUES (1,'Website Redesign Engineering','Engineering','2022-02-10','2022-02-15'),(2,'Product Launch Marketing','Marketing','2021-05-20','2021-10-30'),(3,'Infrastructure Upgrade Engineering','Engineering','2023-01-15','2023-07-30'),(4,'Employee Training Program','HR','2022-10-01','2023-03-31'),(5,'Budget Planning','Finance','2022-12-05','2023-06-30');

--CREATE TABLE Assignments (AssignmentID INTEGER, EmployeeID INTEGER, ProjectID INTEGER, HoursWorked INTEGER);
--INSERT INTO Assignments (AssignmentID , EmployeeID , ProjectID, HoursWorked)
--VALUES (1,1,1,120.5),(2,2,2,90.0),(3,3,1,150.25),(4,4,4,80.75),(5,5,5,110.0);

--SELECT Employees.Name, Assignments.HoursWorked FROM Employees JOIN Assignments ON Employees.EmployeeID = Assignments.EmployeeID;

SELECT Department, AVG(Salary) AS AverageSalary
FROM Employees
GROUP BY Department;
```

**Available Tables**

Employees				
EmployeeID	Name	Department	Salary	HireDate
1	John Doe	Engineering	60000	2022-01-05
2	Alice Smith	Marketing	55000	2021-03-12
3	Bob Johnson	Engineering	65000	2023-05-20
4	Sarah Williams	HR	50000	2003
5	Michael Brown	Finance	70000	2021-11-15

  

Orders			
order_id	item	amount	customer_id
1	Keyboard	400	4
2	Mouse	300	4
3	Monitor	12000	3
4	Keyboard	400	1
5	Mousepad	250	2

**Output**

Department	AverageSalary
Engineering	62500
Finance	70000
HR	50000
Marketing	55000

d) Calculate the maximum hours worked on each project. (3)

The screenshot shows the Online SQL Editor interface. On the left, there's a sidebar with several tables listed:

- Assignments [-]**
  - AssignmentID [integer]
  - EmployeeID [integer]
  - ProjectID [integer]
  - HoursWorked [integer]
- Customers [-]**
  - customer\_id [int]
  - first\_name [varchar(100)]
  - last\_name [varchar(100)]
  - age [int]
  - country [varchar(100)]
- Employees [-]**
  - EmployeeID [integer]
  - Name [varchar(100)]
  - Department [varchar(100)]
  - Salary [integer]
  - HireDate  
[date format 'yyyy-mm-dd']
- Orders [-]**
  - order\_id [integer]
  - item [varchar(100)]

In the main area, there's an **Input** field containing the following SQL code:

```
--CREATE TABLE Assignments (AssignmentID INTEGER, EmployeeID INTEGER, ProjectID INTEGER, Hoursworked INTEGER);
--INSERT INTO Assignments (AssignmentID , EmployeeID , ProjectID, Hoursworked)
--VALUES (1,1,1,120.5),(2,2,2,90.0),(3,3,1,150.25),(4,4,4,80.75),(5,5,5,110.0);

--SELECT Employees.Name, Assignments.Hoursworked FROM Employees JOIN Assignments
ON Employees.EmployeeID = Assignments.EmployeeID;

--SELECT Department, AVG(Salary) AS AverageSalary FROM Employees GROUP BY
Department;

SELECT ProjectID, MAX(Hoursworked) AS MaxHoursworked
FROM Assignments
GROUP BY ProjectID;
```

Below the Input field is an **Output** table:

ProjectID	MaxHoursWorked
1	150.25
2	90
4	80.75
5	110

On the right side, there are three tables:

- Available Tables**: Shows the **Assignments** table.
- Customers**: Shows the following data:

customer_id	first_name	last_name	age	country
1	John	Doe	31	USA
2	Robert	Luna	22	USA
3	David	Robinson	22	UK
4	John	Reinhardt	25	UK
5	Betty	Doe	28	UAE

- Employees**: Shows the following data:

EmployeeID	Name	Department	Salary	HireDate
1	John	Engineering	60000	2022-01-15

e) Retrieve the names of employees in the engineering department. (3)

Online SQL Editor

<https://www.programiz.com/sql/online-compiler/>

**Programiz Online SQL Editor**

**you can save by bundling your home and car.**

**Interactive SQL Course**

**Available Tables**

	3	David	Robinson	22	UK
Employees	4	John	Reinhardt	25	UK
Employees	5	Betty	Doe	28	UAE

**Input**

```
--CREATE TABLE Assignments (AssignmentID INT, EmployeeID INT, ProjectID INT, HoursWorked INT);
--INSERT INTO Assignments (AssignmentID , EmployeeID , ProjectID , HoursWorked)
--VALUES (1,1,1,120.5),(2,2,2,90.0),(3,3,1,150.25),(4,4,4,80.75),(5,5,5,110.0);

--SELECT Employees.Name, Assignments.HoursWorked FROM Employees JOIN Assignments
ON Employees.EmployeeID = Assignments.EmployeeID;

--SELECT Department, AVG(Salary) AS AverageSalary FROM Employees GROUP BY
Department;

--SELECT ProjectID, MAX(HoursWorked) AS MaxHoursWorked FROM Assignments GROUP BY
ProjectID;

SELECT Name
FROM Employees
WHERE Department = 'Engineering';
```

**Output**

Name
John Doe
Bob Johnson

**Employees**

EmployeeID	Name	Department	Salary	HireDate
1	John Doe	Engineering	60000	2022-01-05
2	Alice Smith	Marketing	55000	2021-03-12
3	Bob Johnson	Engineering	65000	2023-05-20
4	Sarah Williams	HR	50000	2003
5	Michael Brown	Finance	70000	2021-11-15

**Orders**

order_id	item	amount	customer_id
1	Keyboard	400	4
2	Mouse	300	4

2°C Cloudy

8:19 PM 2024-02-11

f) Retrieve the names of projects started after 2022. (3)

Online SQL Editor

<https://www.programiz.com/sql/online-compiler/>

**Programiz Online SQL Editor**

**How do top CIOs build AI-enabled enterprises?**

**Interactive SQL Course**

**Available Tables**

	5	Mousepad	250	2
--	---	----------	-----	---

**Projects**

ProjectID	Name	Department	StartDate	EndDate
1	Website Redesign	Engineering	2022-02-10	2022-08-15
2	Product Launch	Marketing	2021-05-20	2021-11-30
3	Infrastructure Upgrade	Engineering	2023-01-15	2023-07-30
4	Employee Training Program	HR	2022-10-01	2023-03-31
5	Budget Planning	Finance	2022-12-05	2023-06-30

**Input**

```
--INSERT INTO Assignments (AssignmentID , EmployeeID , ProjectID , HoursWorked)
--VALUES (1,1,1,120.5),(2,2,2,90.0),(3,3,1,150.25),(4,4,4,80.75),(5,5,5,110.0);

--SELECT Employees.Name, Assignments.HoursWorked FROM Employees JOIN Assignments
ON Employees.EmployeeID = Assignments.EmployeeID;

--SELECT Department, AVG(Salary) AS AverageSalary FROM Employees GROUP BY
Department;

--SELECT ProjectID, MAX(HoursWorked) AS MaxHoursWorked FROM Assignments GROUP BY
ProjectID;

--SELECT Name FROM Employees WHERE Department = 'Engineering';

SELECT Name
FROM Projects
WHERE StartDate > '2022-01-01';
```

**Output**

Name
Website Redesign Engineering
Infrastructure Upgrade Engineering
Employee Training Program
Budget Planning

**Shipments**

shipping_id	status	customer
1	Pending	2

2°C Cloudy

8:24 PM 2024-02-11

g) Show all assignments details where the number of hours worked exceeds 100. (3)

The screenshot shows the Oracle Online SQL Editor interface. On the left, there are four tabs: 'Assignments [-]', 'Customers [-]', 'Employees [-]', and 'Orders [-]'. The 'Assignments [-]' tab is selected, showing its schema: AssignmentID [integer], EmployeeID [integer], ProjectID [integer], and HoursWorked [integer]. The 'Input' section contains the following SQL code:

```
--SELECT Employees.Name, Assignments.HoursWorked FROM Employees JOIN Assignments  
ON Employees.EmployeeID = Assignments.EmployeeID;  
  
--SELECT Department, AVG(Salary) AS AverageSalary FROM Employees GROUP BY  
Department;  
  
--SELECT ProjectID, MAX(HoursWorked) AS MaxHoursWorked FROM Assignments GROUP BY  
ProjectID;  
  
--SELECT Name FROM Employees WHERE Department = 'Engineering';  
  
--SELECT Name FROM Projects WHERE StartDate > '2022-01-01';  
  
SELECT *  
FROM Assignments  
WHERE HoursWorked > 100;
```

The 'Output' section displays the results of the last query:

AssignmentID	EmployeeID	ProjectID	HoursWorked
1	1	1	120.5
3	3	1	150.25
5	5	5	110

The 'Available Tables' section lists three tables: 'Assignments', 'Customers', and 'Employees'. The 'Assignments' table has the following data:

AssignmentID	EmployeeID	ProjectID	HoursWorked
1	1	1	120.5
2	2	2	90
3	3	1	150.25
4	4	4	80.75
5	5	5	110

The 'Customers' table has the following data:

customer_id	first_name	last_name	age	country
1	John	Doe	31	USA
2	Robert	Luna	22	USA
3	David	Robinson	22	UK
4	John	Reinhardt	25	UK
5	Betty	Doe	28	UAE

The 'Employees' table has the following data:

EmployeeID	Name	Department	Salary	HireDate
1	John	Engineering	60000	2022-01-

The bottom of the screen shows a taskbar with various icons and system status information: 2°C Cloudy, Search, File, Home, Help, ENG US, 8:27 PM, 2024-02-11.

h) Retrieve employee name along with the name of project assigned and the hours worked on each project. (3)

Online SQL Editor

How do top CIOs build AI-enabled enterprises? Get the ebook

Interactive SQL Course

**Input**

```
--SELECT ProjectID, MAX(HoursWorked) AS MaxHoursworked FROM Assignments GROUP BY ProjectID;

--SELECT Name FROM Employees WHERE Department = 'Engineering';

--SELECT Name FROM Projects WHERE StartDate > '2022-01-01';

--SELECT * FROM Assignments WHERE HoursWorked > 100;

SELECT Employees.Name AS EmployeeName, Projects.Name AS ProjectName,
Assignments.Hoursworked
FROM Employees
JOIN Assignments ON Employees.EmployeeID = Assignments.EmployeeID
JOIN Projects ON Assignments.ProjectID = Projects.ProjectID;
```

**Available Tables**

2	Mouse	300	4	
3	Monitor	12000	3	
4	Keyboard	400	1	
5	Mousepad	250	2	

**Projects**

ProjectID	Name	Department	StartDate	EndDate
1	Website Redesign Engineering	Engineering	2022-02-10	2022-08-15
2	Product Launch Marketing	Marketing	2021-05-20	2021-11-30
3	Infrastructure Upgrade Engineering	Engineering	2023-01-15	2023-07-30
4	Employee Training Program	HR	2022-10-01	2023-03-31
5	Budget Planning	Finance	2022-12-05	2023-06-30

**Output**

EmployeeName	ProjectName	HoursWorked
John Doe	Website Redesign Engineering	120.5
Alice Smith	Product Launch Marketing	90
Bob Johnson	Website Redesign Engineering	150.25
Sarah Williams	Employee Training Program	80.75
Michael Brown	Budget Planning	110

Cloudy 2°C ENG US 8:38 PM 2024-02-11

i) Who are the top 3 employees with the highest salary? (3)

Online SQL Editor

Reach over 3,500 hard-to-access private... SPONSORED BY HOMESPHERE LEARN MORE Interactive SQL Course

**Input**

```
--SELECT Name FROM Employees WHERE Department = 'Engineering';

--SELECT Name FROM Projects WHERE StartDate > '2022-01-01';

--SELECT * FROM Assignments WHERE HoursWorked > 100;

--SELECT Employees.Name AS EmployeeName, Projects.Name AS ProjectName,
Assignments.Hoursworked FROM Employees
--JOIN Assignments ON Employees.EmployeeID = Assignments.EmployeeID
--JOIN Projects ON Assignments.ProjectID = Projects.ProjectID;

SELECT Name, Salary
FROM Employees
ORDER BY Salary DESC
LIMIT 3;
```

**Available Tables**

2	Mouse	300	4	
3	Monitor	12000	3	
4	Keyboard	400	1	
5	Mousepad	250	2	

**Projects**

ProjectID	Name	Department	StartDate	EndDate
1	Website Redesign Engineering	Engineering	2022-02-10	2022-08-15
2	Product Launch Marketing	Marketing	2021-05-20	2021-11-30
3	Infrastructure Upgrade Engineering	Engineering	2023-01-15	2023-07-30
4	Employee Training Program	HR	2022-10-01	2023-03-31
5	Budget Planning	Finance	2022-12-05	2023-06-30

**Output**

Name	Salary
Michael Brown	70000
Bob Johnson	65000
John Doe	60000

Cloudy 2°C ENG US 8:41 PM 2024-02-11

- j) Retrieve the names of employees who have not been assigned to projects starting with the letter 'P' and have worked more than 100 hours. (3)

The screenshot shows the Online SQL Editor interface on Programiz. The left sidebar lists three tables: Assignments, Customers, and Employees. The main area has an 'Input' section containing the following SQL code:

```
--SELECT * FROM Assignments WHERE HoursWorked > 100;

--SELECT Employees.Name AS EmployeeName, Projects.Name AS ProjectName,
Assignments.HoursWorked FROM Employees
--JOIN Assignments ON Employees.EmployeeID = Assignments.EmployeeID
--JOIN Projects ON Assignments.ProjectID = Projects.ProjectID;

--SELECT Name, Salary FROM Employees ORDER BY Salary DESC LIMIT 3;

SELECT e.Name
FROM Employees e
LEFT JOIN Assignments a ON e.EmployeeID = a.EmployeeID
LEFT JOIN Projects p ON a.ProjectID = p.ProjectID
WHERE (p.Name IS NULL OR p.Name NOT LIKE 'P%')
AND (a.HoursWorked > 100 OR a.HoursWorked IS NULL);
```

The 'Output' section displays the results of the query, listing employee names: John Doe, Bob Johnson, and Michael Brown.

The right side shows the data for the three tables:

- Assignments** table:

AssignmentID	EmployeeID	ProjectID	HoursWorked
1	1	1	120.5
2	2	2	90
3	3	1	150.25
4	4	4	80.75
5	5	5	110

- Customers** table:

customer_id	first_name	last_name	age	country
1	John	Doe	31	USA
2	Robert	Luna	22	USA
3	David	Robinson	22	UK
4	John	Reinhardt	25	UK
5	Betty	Doe	28	UAE

- Employees** table:

customer_id	first_name	last_name	age	country
1	John	Doe	31	USA
2	Robert	Luna	22	USA
3	David	Robinson	22	UK
4	John	Reinhardt	25	UK
5	Betty	Doe	28	UAE

- k) Retrieve the department names where the average salary is above the company-wide average salary. (3)

Online SQL Editor <https://www.programiz.com/sql/online-compiler/>

**Programiz** Online SQL Editor

**ORACLE** How do top CIOs build AI-enabled enterprises? Get the ebook

**Interactive SQL Course**

**Assignments [-]**

- AssignmentID [integer]
- EmployeeID [integer]
- ProjectID [integer]
- HoursWorked [integer]

**Customers [-]**

- customer\_id [int]
- first\_name [varchar(100)]
- last\_name [varchar(100)]
- age [int]
- country [varchar(100)]

**Employees [-]**

- EmployeeID [integer]
- Name [varchar(10)]
- Department [varchar(10)]
- Salary [integer]
- HireDate  
[date format 'yyyy-mm-dd']

**Orders [-]**

- order\_id [integer]
- item [varchar(100)]

**Input**

```
--ASSIGNMENTS.HOURSWORKED > FROM Employees
--JOIN Assignments ON Employees.EmployeeID = Assignments.EmployeeID
--JOIN Projects ON Assignments.ProjectID = Projects.ProjectID;
--SELECT Name, Salary FROM Employees ORDER BY Salary DESC LIMIT 3;

--SELECT e.Name FROM Employees e
--LEFT JOIN Assignments a ON e.EmployeeID = a.EmployeeID
--LEFT JOIN Projects p ON a.ProjectID = p.ProjectID
--WHERE (p.Name IS NULL OR p.Name NOT LIKE 'P%') AND (a.HoursWorked > 100 OR
a.HoursWorked IS NULL);

SELECT Department
FROM Employees
GROUP BY Department
HAVING AVG(Salary) > (SELECT AVG(Salary)FROM Employees);
```

**Available Tables**

	3	David	Robinson	22	UK
	4	John	Reinhardt	25	UK
	5	Betty	Doe	28	UAE

**Employees**

EmployeeID	Name	Department	Salary	HireDate
1	John Doe	Engineering	60000	2022-01-05
2	Alice Smith	Marketing	55000	2021-03-12
3	Bob Johnson	Engineering	65000	2023-05-20
4	Sarah Williams	HR	50000	2003
5	Michael Brown	Finance	70000	2021-11-15

**Output**

Department
Engineering
Finance

**Orders**

order_id	item	amount	customer_id
1	Keyboard	400	4
2	Mouse	300	4

- I) Retrieve employee names who have worked on projects that started after 2022 and ended before 2024. (3)

The screenshot shows the Oracle Online SQL Editor interface. On the left, there's a sidebar with tables: Assignments, Customers, Employees, and Orders. The main area has an 'Input' tab where a complex SQL query is written, and an 'Output' tab showing the results. The 'Available Tables' section lists the same four tables. To the right, there are two tables: 'Employees' and 'Orders'. The 'Employees' table has data for 5 employees: John Doe, Alice Smith, Bob Johnson, Sarah Williams, and Michael Brown. The 'Orders' table has data for 3 orders.

```

--SELECT Name, Salary FROM Employees ORDER BY Salary DESC LIMIT 3;

--SELECT e.Name FROM Employees
--LEFT JOIN Assignments a ON e.EmployeeID = a.EmployeeID
--LEFT JOIN Projects p ON a.ProjectID = p.ProjectID
--WHERE (p.Name IS NULL OR p.Name NOT LIKE 'P%') AND (a.HoursWorked > 100 OR
a.HoursWorked IS NULL);

--SELECT Department FROM Employees GROUP BY Department
--HAVING AVG(Salary) > (SELECT AVG(Salary)FROM Employees);

SELECT DISTINCT e.Name
FROM Employees e
JOIN Assignments a ON e.EmployeeID = a.EmployeeID
JOIN Projects p ON a.ProjectID = p.ProjectID
WHERE p.StartDate > '2022-01-01' AND p.EndDate < '2024-01-01';

```

EmployeeID	Name	Department	Salary	HireDate
1	John Doe	Engineering	60000	2022-01-05
2	Alice Smith	Marketing	55000	2021-03-12
3	Bob Johnson	Engineering	65000	2023-05-20
4	Sarah Williams	HR	50000	2003
5	Michael Brown	Finance	70000	2021-11-15

order_id	item	amount	customer_id
1	Item A	100	1
2	Item B	200	2
3	Item C	150	3

- m) Retrieve the names of employees whose names contain the letter 'a' and have worked on projects where the project name starts with 'P'. Display their names along with the total number of hours worked on those projects. (4)

Online SQL Editor

https://www.programiz.com/sql/online-compiler/

**Programiz**

Online SQL Editor

**Interactive SQL Course**

**Input**

```
--WHERE (p.Name IS NULL OR p.Name NOT LIKE '%P') AND (a.HoursWorked > 100 OR a.Hoursworked IS NULL);

--SELECT Department FROM Employees GROUP BY Department
--HAVING AVG(Salary) > (SELECT AVG(Salary)FROM Employees);

--SELECT DISTINCT e.Name FROM Employees
--JOIN Assignments a ON e.EmployeeID = a.EmployeeID
--JOIN Projects p ON a.ProjectID = p.ProjectID
--WHERE p.StartDate > '2022-01-01' AND p.EndDate < '2024-01-01';

SELECT e.Name, SUM(a.Hoursworked) AS TotalHoursWorked
FROM Employees e
JOIN Assignments a ON e.EmployeeID = a.EmployeeID
JOIN Projects p ON a.ProjectID = p.ProjectID
WHERE e.Name LIKE '%A%' AND p.Name LIKE '%P%'
GROUP BY e.Name;
```

**Available Tables**

	3	David	Robinson	22	UK
Employees	4	John	Reinhardt	25	UK
Orders	5	Betty	Doe	28	UAE

**Employees**

EmployeeID	Name	Department	Salary	HireDate
1	John Doe	Engineering	60000	2022-01-05
2	Alice Smith	Marketing	55000	2021-03-12
3	Bob Johnson	Engineering	65000	2023-05-20
4	Sarah Williams	HR	50000	2003
5	Michael Brown	Finance	70000	2021-11-15

**Output**

Name	TotalHoursWorked
Alice Smith	90

**Orders**

order_id	item	amount	customer_id
1	Keyboard	400	4
2	Mouse	300	4

Cloudy 1°C

Search

9:23 PM 2024-02-11