

REPORT:

sales for each employee:

1/John Doe: 49588

2/Jane Smith: 74641

3/Alice Brown: 55560

4/Charlie Lee: 47589

5/Bob Johnson: 66747

Top3 employees in terms of sales:

1/John Doe

2/Jane Smith

3/Alice Brown

The department with the highest total salaries is the department with 102 as ID
And the total salaries is 125000.00

List each employee and their supervisor's ID:

EmployeeID | EMPLOYEE NAME | SupervisorID

1	Cheyenne Padilla	
2	Michael Martin	6
3	Tim Wright	5
4	Kristy Archer	1
1 5	Robert Rios	3
6	Gregory Casey	3
7	Douglas Huber	3
8	Bobby Browning	8
9	Crystal Wilson	7
10	Tammy Adams	2

customers with purchases above \$10,000:

no customers made **above \$10,000.**

SQL CODE:

```
CREATE TABLE joined_table AS
SELECT emp.EmployeeID, name, salary, DepartmentID, SupervisorID, PurchaseDate, SalesAmount,
TransactionID
FROM emp
JOIN sales ON emp.EmployeeID = sales.EmployeeID;
select name, sum(salesamount) from joined_table group by Name; -- Retrieve the total sales
for each employee
-- Identify the top-performing employee in terms of sales
SELECT name, SUM(salesamount) AS TotalSales
FROM joined_table
GROUP BY name
ORDER BY TotalSales DESC
LIMIT 1;
-- Find departments with the highest total salaries.
select name, DepartmentID, sum(salary) as totalsalary
from joined_table
group by DepartmentID
order by totalsalary desc;
-- List each employee and their supervisor's name.
select name, supervisorid from joined_table group by SupervisorID;
-- Retrieve customers with purchases above $10,000.
select customerid, sum(salesamount) as totalpurchase from sales
group by EmployeeID
having totalpurchase > 10000;
```

PYTHON CODE:

```
import pandas as pd
# Load the Data
data = pd.read_csv('sales_data.csv')
sales = pd.DataFrame(data)
data = pd.read_csv('employee_data.csv')
emp = pd.DataFrame(data)
#Clean Sales Data
sales['SalesAmount'] = sales['SalesAmount'].replace({r'\$': '', ',': ''},
regex=True).astype(float)
sales['SalesAmount']=pd.to_numeric(sales['SalesAmount'], errors= 'coerce')
sales['SalesAmount'].dtype
sales['PurchaseDate'] = pd.to_datetime(sales['PurchaseDate'], errors='coerce')
sales.fillna({'PurchaseDate': 'NaT'}, inplace=True)
print(sales.dtypes)
#Clean employee Data
emp['Salary'] = emp['Salary'].replace({r'\$': '', ',': ''}, regex=True).astype(int)
emp['Salary'].dtype
emp['Salary']=pd.to_numeric(emp['Salary'], errors= 'coerce')
emp['SupervisorID']=pd.to_numeric(emp['SupervisorID'], errors = 'coerce')
emp.fillna({'SupervisorID': 0}, inplace=True)
emp['SupervisorID'] = emp['SupervisorID'].astype(int)
print(emp)
print(emp.dtypes)
#Save Cleaned Data
task_cleaned_data = emp.copy()
task_cleaned_data.to_csv('emp.csv', index=False)
task_cleaned_data = sales.copy()
task_cleaned_data.to_csv('sales.csv', index=False)
```