Queries for Scenario

Scenario 1: a) Identify vendors who have been paid and have valid receipt records to ensure accurate financial tracking and policy compliance.

```
SELECT VendorName, VendorCategory, InvoiceNo, Amount AS PaidAmount, TotalAmt As ReceiptTotal
FROM Travel_Vendor v, Expense_Record er, Payment_Record pr, Receipts r
WHERE v.VendorID = er.VendorID AND er.ExpenseID = pr.ExpenseID AND er.ExpenseID = r.ExpenseID
```

VendorName	VendorCate	InvoiceNo	PaidAmount	ReceiptTota
SkyHigh Airlines	Airline	INV-3822	$525.00	$525.00
World Car Service	Taxi	INV-5167	$430.00	$430.00
JetLink Airlines	Airline	INV-9003	$881.00	$881.00
GoCar Rentals	Car	INV-3886	$1,351.00	$1,351.00
TravelSecure	Other	INV-8505	$745.00	$745.00

b) Identify vendors with recorded expenses that have not yet been paid, to track pending payments and manage outstanding liabilities.

```
SELECT  VendorName, InvoiceNo, er.ExpenseAmt
FROM (Travel_Vendor  v INNER JOIN Expense_Record  er ON v.VendorID = er.VendorID)
LEFT OUTER JOIN Payment_Record  pr ON er.ExpenseID = pr.ExpenseID
WHERE pr.PaymentNo IS NULL
```

VendorName	InvoiceNo	ExpenseAmt
GrandView Hotels	INV-6747	1481.00
GreatView Air	INV-7486	806.00
SkyHigh Airlines	INV-3822	1025.00
*		

Scenario 2: **Identify expenses that exceed their category's maximum allowed amount to detect policy violations and prevent overspending.**

```
SELECT e.ExpenseID, e.ExpenseAmt, c.CategoryMaxLimit
FROM Expense_Record e
INNER JOIN ExpenseCategory c ON e.ExpenseCategory = c.ExpenseCategory
```

```
WHERE e.ExpenseAmt > (
        SELECT CategoryMaxLimit
        FROM ExpenseCategory
        WHERE ExpenseCategory = e.ExpenseCategory
);
```

ExpenseID	ExpenseAmt	CategoryMa
X3000	1750.00	$1,500.00
X3001	330.00	$300.00
X3002	881.00	$1,500.00
X3004	745.00	$600.00
X3006	806.00	$1,500.00

Scenario 3: **List the names of employees, the travel request ID, and total expense amount only for those travel requests whose total expenses exceed $800.**

```
SELECT e.StaffName, tr.TravelRequestID, (SELECT SUM(er.ExpenseAmt)
FROM Expense_Record AS er
WHERE er.TravelRequestID = tr.TravelRequestID) AS TotalExpense
FROM Employee AS e
INNER JOIN Travel_Request AS tr ON e.StaffID = tr.StaffID
WHERE (SELECT SUM(er.ExpenseAmt)
FROM Expense_Record AS er
WHERE er.TravelRequestID = tr.TravelRequestID) > 800
```

Query1

StaffName	TravelRequestID	TotalExpense
Kristen Gomez	TR4003	1351
Hector Dalton	TR4000	1750
Mariela Schultz	TR4002	881
Angela Molina	TR4005	1481
Gerald Preston	TR4006	806

Scenario 4: Identify all departments located in cities that start with “S.”

```
SELECT *
FROM department
WHERE location LIKE 'S*'
```

DepartmentI	DeptName	Location
1003	Sales	San Francisco
1005	HR	Seattle

Scenario 5: List employees who have submitted more than one travel request to monitor frequent travel and optimize resource allocation.

```
SELECT StaffName, COUNT(TravelRequestId) AS NumberOfRequests
FROM Staff s INNER JOIN Travel_Request tr
ON s.StaffID = tr.StaffID
GROUP BY StaffName
HAVING COUNT(TravelRequestId) > 1;
```

StaffName	NumberOfRequests
Mariela Schultz	2

Scenario 6: Show the number of days it took to upload digital copies of receipts in the system.

```
SELECT  e.ExpenseID,  e.ExpenseDate, r.UploadDate,  DateDiff("d", e.ExpenseDate, r.UploadDate) AS
DaysToUpload
FROM Expense_Record AS e
INNER JOIN Receipts AS r ON e.ExpenseID = r.ExpenseID
ORDER BY DateDiff("d", e.ExpenseDate, r.UploadDate) DESC;
```

ExpenseID	ExpenseDat	UploadDate	DaysToUplo
X3004	3/9/2025	3/10/2025	1
X3003	3/8/2025	3/9/2025	1
X3002	3/29/2025	3/30/2025	1
X3001	4/8/2025	4/9/2025	1
X3000	3/19/2025	3/20/2025	1

Scenario 7: Find all payments made by a credit card; this can help auditors find out how many times the employees use their company card.

```
SELECT * FROM Payment_Record
WHERE PaymentType = "Credit Card";
```

PaymentNo	Amount	PaymentRec	PaymentTyp	ExpenseID
P6001	$430.00	4/10/2025	Credit Card	X3001
P6003	$1,351.00	3/10/2025	Credit Card	X3003

Scenario 8: Identify the highest receipt amount to flag for audit review, ensuring spending stays within policy limits.

```
SELECT TOP 1
FORMAT(TotalAmt, '$#,##0.00') AS MaxReceiptAmount, ReceiptNo, ExpenseID
FROM Receipts
ORDER BY TotalAmt DESC;
```

MaxReceipt	ReceiptNo	ExpenseID
$1,351.00	R5003	X3003

Scenario 9: Identify the highest receipt amount to flag for audit review, ensuring spending stays within policy limits.

```
SELECT e.StaffName, 'No Expenses' AS ExpenseStatus
FROM Employee e
WHERE e.StaffID NOT IN (SELECT tr.StaffID FROM Travel_request tr INNER JOIN Expense_Record er ON
tr.TravelRequestId = er.TravelRequestID)
UNION
SELECT e.StaffName, 'Has Expenses' AS ExpenseStatus
FROM Employee e
WHERE e.StaffID IN (SELECT tr.StaffID FROM Travel_request tr INNER JOIN Expense_Record er ON
tr.TravelRequestId = er.TravelRequestID)
```

StaffName	ExpenseStatus
Angela Molina	Has Expenses
Carlee Yu	No Expenses
Gerald Preston	Has Expenses
Hector Dalton	Has Expenses
Kristen Gomez	Has Expenses
Mariela Schultz	Has Expenses
Reid Stein	Has Expenses

Scenario 10: List all vendors in the Flight, Hotel, or Transportation categories along with their contact emails for communication and record-keeping.

```
SELECT VendorName, VendorCategory, VendorEmail
FROM Travel_Vendor
WHERE VendorCategory
IN ('Flight', 'Hotel', 'Transportation')
```

VendorName	VendorCate	VendorEmail
GrandView Hotels	Hotel	reservations@grandview.com

Scenario 11: List the top 3 employees who submitted travel requests with the highest total estimated costs. Include the employee's name, department, and their total estimated travel costs.

```
SELECT TOP 3  s.StaffName, d.DeptName, SUM(tr.EstimatedCost) AS TotalEstimatedCost
FROM (Travel_Request AS tr INNER JOIN Staff AS s ON tr.StaffID = s.StaffID)
INNER JOIN Department AS d ON s.DepartmentID = d.DepartmentID
GROUP BY s.StaffName, d.DeptName
ORDER BY SUM(tr.EstimatedCost) DESC
```

StaffName	DeptName	TotalEstimatedCost
Mariela Schultz	Marketing	4324
Kristen Gomez	Marketing	2679
Angela Molina	Sales	2544

Scenario 12: Retrieve all approval requests submitted after April 21, 2025, ordered by the approval date.

```
SELECT *
FROM approval_track
WHERE ApprovalDate > #04/21/2025#
```

ORDER BY ApprovalDate;

ApprovalID	ApprovalSta	ApprovalDat	ApprovalComments	StaffID
A8000	Approved	4/26/2025	Let finance know if plans change.	E3442
A8005	Approved	4/28/2025	Safe travels and please follow policy.	E3445
A8007	Pending	4/29/2025	Staying within budget is a concern.	E3451
A8003	Rejected	5/2/2025	Please submit receipts.	E3450
A8001	Pending	5/3/2025	Identify key meeting dates.	E3450

Scenario 13: Find the earliest and latest payment dates recorded in the system to analyze the payment timeline.

SELECT MIN(PaymentRecordDate) AS FirstPayment, MAX(PaymentRecordDate) AS LastPayment
FROM Payment_Record;

FirstPayment	LastPayment
3/10/2025	4/14/2025