Answer Code

Query 1: Display the Branch Name and City of Deployment for Active Assets in Inactive Branches

```
SELECT
    asset_id,
    asset_name,
    asset_status,
    Branch_ID,
    Branch_name,
    Branch_city,
    Branch_status,
    asset_cost
FROM Assets
LEFT JOIN Branch_Data
ON assets.deployed_branch_id = Branch_Data.Branch_ID
WHERE Branch_status = 'Inactive'
AND asset_status !='Inactive';
```

- **Fields Selected**:
  - `asset_id`, `asset_name`, `asset_status`, `asset_cost`: Information related to each asset.
  - `Branch_ID`, `Branch_name`, `Branch_city`, `Branch_status`: Information about the branch where the asset is deployed.
- `LEFT JOIN`: The `Assets` table is joined with the `Branch_Data` table using the `deployed_branch_id` to match assets with the branch details.
- `WHERE` Clause:
  - `Branch_status = 'Inactive'`: Filters out only the branches that are marked as "Inactive."
  - `asset_status != 'Inactive'`: Ensures only active assets (those not marked as "Inactive") are selected.

- **Result**: The output will display the `Branch_name` and `Branch_city` along with other details for each active asset that is deployed in an inactive branch.

Reasoning:

This query is useful for identifying assets that are still active but are deployed in branches that are no longer operational. It helps in assessing whether these assets need to be relocated, repurposed, or dealt with in some other way.

Query 2: Calculate the Cost of Active Assets Deployed in Inactive Branches

```
SELECT
   SUM(asset_cost) as 'cost_at_inactive_branches'
FROM Assets
LEFT JOIN Branch_Data
ON assets.deployed_branch_id = Branch_Data.Branch_ID
WHERE Branch_status = 'Inactive'
AND asset_status !='Inactive';
```

Explanation:

- **Purpose**: This query calculates the total cost of assets that are still active (not marked as "Inactive") but are deployed in branches that are marked as "Inactive."
- **SUM(asset_cost)**: This function sums up the cost (`asset_cost`) of all such assets.
- `LEFT JOIN`: Similar to the first query, this join matches assets with their corresponding branch details.
- `WHERE` **Clause**:
  - `Branch_status = 'Inactive'`: Filters out only the inactive branches.
  - `asset_status != 'Inactive'`: Ensures only the active assets are considered.
- **Result**: The output will be a single value (`cost_at_inactive_branches`) representing the total cost of all active assets in inactive branches.

Reasoning:

This query is designed to provide a financial overview of the resources that are tied up in inactive branches. It helps in understanding the monetary value of assets that may need to be re-evaluated for efficiency or redeployment.

Summary:

- **Query 1** identifies the active assets still deployed in inactive branches, displaying detailed information about the assets and their corresponding branches.
- **Query 2** calculates the total financial value of these active assets in inactive branches, giving insight into the potential cost implications.