Instructions:
Please code the solution exclusively in python, in a single file. Provide an answer at your earliest convenience, no later than 1 week from receipt.
You are welcome to send any questions to sfee@banjohealth.com

Problem:

You are given a rule-defined decision tree in a table format. You must find all possible paths that can be achieved. A path consists of an ordered list of nodes where the ultimate value is a decision {end_approve or end_deny}.

Assumptions:
All Node IDs can be assumed to be unique.
All Nodes must have binary endpoints.

Output:

There should be an ValueError if the path is circular (ie: a single node appears multiple times within a single path and would enable an infinitely looping path)

You should write a function **generate_paths**(df_in: pd.DataFrame) -> List[List[ids]] that takes in a pandas dataframe in the above format.

The output should be a list of paths where each path is a list of nodes that it went through [Node_ID1, NodeID2, …, Decision]

You can assume that the node that appears in the first row is the starting node.

Example:
Example table input

Node_ID	**IF_TRUE**	**IF_FALSE**
1	2	3
2	End_approve	3
3	End_deny	End_approve

Expected Output: [[1, 2, approve], [1, 2, 3, deny], [1, 3, deny], [1, 2, 3, approve], [1, 3, approve]]