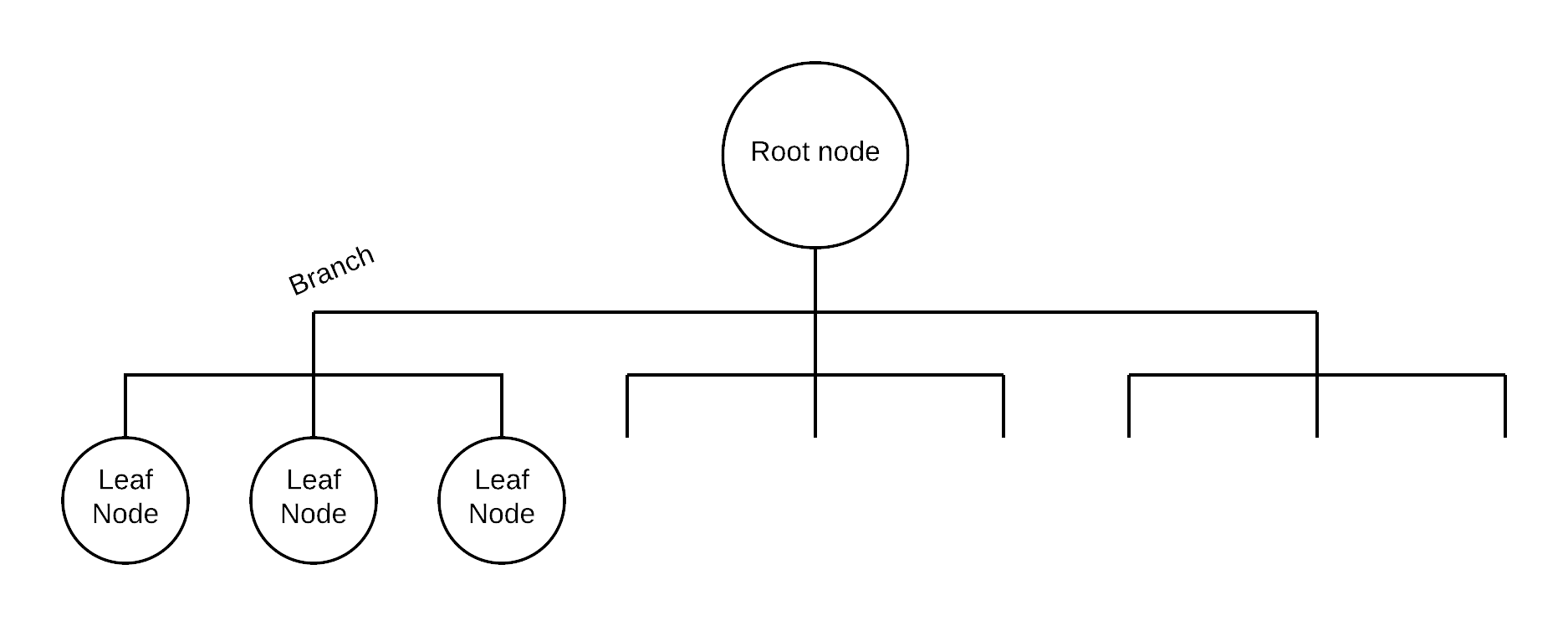
**Diagram a Balanced Tree Index**



As more data is added to a balance tree, the structure of the tree gets wider, not deeper. There will only ever be two steps to get from the root to a leaf (unless the index is insanely large, then there will be two branch levels).

**What is the difference between a clustered and non-clustered index? How do they work together?**

Only one clustered index can exist on any table, and is most effective when indexed on small, sequential data, such as the primary key or a timestamp. These kinds of indexes work best in write-centric environments.

Many non-clustered indexes can exist any table, and they work best when the index is created from columns used in the WHERE clauses of queries. Non-clustered indexes are optimized for read-centric environments, and while both kinds of indexes harm write performance, non-clustered indexes perform worse for writing.

The only difference in the structure of the balanced tree for clustered and non-clustered indexes is in the lead node. Clustered index leaf nodes point to data in a table, while non-clustered index leaf nodes point to the root of a clustered index. By working together in such a way, the database is able to avoid computing more indexes than necessary by making use of already existing clustered indexes, leading to increased read performance while minimizing the write penalty.