Solutions to Gradiance Lab 1 – Kings

The kings and queens of England are listed in a relation Kings(name,nickname,house,beginReign,endReign). Their name is unique, either using a Roman numeral to distinguish them, e.g., 'Edward I' or 'Edward II', or in a few cases using their nickname, e.g., 'Edward the Confessor'. The attribute `nickname` is an additional appelation, if they have one and it is not used in their name to distinguish them, e.g., 'The Unready'. The value of `nickname` is NULL if there is no nickname. The attribute `house` is the dynasty, e.g., 'Tudor'. Attribute `beginReign` and `endReign` are integers, the first and last years, respectively, that the king or queen was on the throne.

There is also a relation Parents(child,parent). Both attributes are the names of kings or queens, with the obvious connection that the first is a child of the second. Write the following queries:

1. **Who was king in the year 1000? Give the name and nickname.**
2. **Find all the pairs of kings or queens (A,B) such that A was the great grandchild of B.**
3. **Find the name and nickname of all kings or queens that have a nickname that does not begin with "The".**
4. **Find the names of those kings or queens that were the parent of two or more kings or queens. List each such person only once.**
5. **Find for each house the number of kings or queens of that house.**
6. **Several times in British history, kings or queens have deposed one another, so that their reigns overlapped. Find all such pairs, listing the pairs in both orders; i.e., list both (A,B) and (B,A). However, be careful not to list pairs A and B where the only overlap is**

that A's reign ended in the same year that B's began, or vice-versa.

---

1. **SELECT name, nickname FROM Kings WHERE beginReign <=1000 AND endReign >=1000**
2. **SELECT p1.child, p3.parent FROM Parents p1, Parents p2, Parents p3 WHERE p1.parent = p2.child AND p2.parent = p3.child**
3. **SELECT name, nickname FROM Kings WHERE nickname NOT LIKE 'The%'**
4. **SELECT parent FROM Parents GROUP by parent HAVING COUNT(*) > 1**
5. **SELECT house, COUNT(*) FROM Kings GROUP BY house**
6. **SELECT k1.name, k2.name FROM Kings k1, Kings k2 WHERE k1.name <> k2.name AND k1.beginReign < k2.endReign AND k2.beginReign < k1.endReign**