Exercise 8.11 Consider the following relations:

Emp(eid: integer, ename: varchar, sal: integer, age: integer, did: integer)
Dept(did: integer, budget: integer, floor: integer, mgr_eid: integer)

Salaries range from $10,000 to $100,000, ages vary from 20 to 80, each department has about five employees on average, there are 10 floors, and budgets vary from $10,000 to $1 million. You can assume uniform distributions of values.

For each of the following queries, which of the listed index choices would you choose to speed up the query? If your database system does not consider index-only plans (i.e., data records are always retrieved even if enough information is available in the index entry), how would your answer change? Explain briefly.

1. Query: Print ename, age, and sal for all employees.
   a. Clustered hash index on <ename, age, sal> fields of Emp.
   b. Unclustered hash index on <ename, age, sal> fields of Emp. - **get details from emp , unclustered fits better in this scenario**
   c. Clustered B+ tree index on <ename, age, sal> fields of Emp.
   d. Unclustered hash index on <eid, did> fields of Emp.
   e. No index. - **indexing not required, will go through all employee records**
2. Query: Find the dids of departments that are on the 10th floor and have a budget of less than $15,000.
   a. Clustered hash index on the floor field of Dept.
   b. Unclustered hash index on the floor field of Dept.
   c. Clustered B+ tree index on <floor, budget> fields of Dept. - **data sequentially on the two fields, so use clustered**
   d. Clustered B+ tree index on the budget field of Dept.
   e. No index.