Consider the table *Drug* in the Drug Ordering databasementioned in Question 2, and answer the following questions.

*Drug*(*drug\_NO*, *name*, *type*, *production\_date*, *expiration\_date*, *unit*)

(1) It is assumed that there are 20,000 tuples in the table *Drug* that is stored in an on-disk database file named *drugfile*. The file consists of disk blocks and each block contains 20 tuples. Nether a primary or candidate key nor a clustering or non-clustering index is created on the table. For the following query

select *drug\_NO*, *name*, *expiration\_date*

from *Drug*

where *drug\_NO*=*K*

, where *K* is a valid value of *drug\_NO* appearing in *drugfile*, what is the average number of blocks transferred from disk to main memory to execute this query, and why ?

(2) The attribute *drug\_NO* is defined as the primary key and a clustering index is defined on it, and all tuples is then imported to the table. Afterwards, another index *NameInd* is defined on the attribute *name*.

Is *NameInd* a clustering index or non-clustering index, and why ?

(3) For the following SQL query, in addition to the primary index on the primary key *drug\_NO*, on which attributes the indices can be further defined to speed up the query?

select *type*, count(*drug\_NO*)

from *Drug*

where *production\_date* < 2020-01-01

group by *type*

(4) For the following *insert* operation

insert into *Drug*(*drug\_NO*, *name*, *type*, *production\_date*, *expiration\_date*, *unit*)

values(3899, TuiSh, fever, ChineseMed, 2020-01-01, 2023-01-01, mg)

, does the index *NameInd* speed up the operation, and why?