Here are some relations that exist in a database for an orchestra.

Person(email, name, age)

- This relation stores anyone who has signed up for our mailing list. Tuples in this relation may not be listed in Purchase.

Show(id, year, month, day, showing, attendanceNumber)

- Showing describes whether a show was during morning, afternoon, or evening
- {year, month, day, showing} is a candidate key for Show

Song(composer, title)

SongsPerformed(**showID**, **composer**, **title**)

- showID is a foreign key referring to Show
- composer and title are foreign keys referring to attributes of the same name in Song

Purchase(**email**, **showID**, price)

- email is a foreign key referring to the email attribute in Person
- showID is a foreign key referring to Show

Musician(id, name, instrument, position, nationality)

PerformedIn(**id**, **showID**)

- id refers to the attribute of the same name in Musician
- showID is a foreign key referring to Show

Write SQL statements to answer the following questions:

1. Use the INTERSECT operator for the following question.

   Find the showIDs of shows where the symphony performed songs by Mozart and Beethoven.

   Another way to think about this question: Find the shows where at least one song composed by Mozart and at least one song composed by Beethoven were performed.

2. Write a query to solve question 1 but this time, do not use the INTERSECT query. If this is not possible, explain why.

3. Write a query to solve question 1 with the EXISTS/NOT EXISTS operator. If this is not possible, explain why.

4. Write a query to solve question 1 with the IN/NOT IN operator. If this is not possible, explain why.