Exercise 1

Alternative to school data creation: it would be better to delegate the responsibility of setting the school data to the School class. As this is a small project with the results printed by console, this part is at the Main class.

C) I use a subject id assuming there would be an API request providing that id.

- Another option would be to redefine the Subject equals method and send a whole Subject (to the School method that resolves the exercise) and search by it in the student's subject list.

D)

E) The query returns all the columns from janitor, employee and person tables.

- As we only need the first and last name we can replace the * from the select clause with eg. (p.person_name, p.person_lastname)

- Another option can be to create a non-clustered index with the janitor working area field (ordered by it). Without an index, the DB engine has to make a table scan because by default the table is ordered by its id, so it has to check all rows to compare the working area name with the one given in the query.
  A non-clustered index, use of a B-tree structure that allows it to find logarithmically the rows with the working area given in the query. This makes an important improvement of the query performance.

- In case the query delay is caused by the huge quantity of rows of the result, we can imagine there is a user application requesting data and does not need to show all the results together. In that way we can implement a system pagination where instead of returning all the resulting rows, we divide the result into pages with less data and when the user asks for another page we repeat the same process to obtain other rows. This can be done by using some of the sql clauses like TOP or LIMIT that limit the number of rows to return.

F) I would use a non-relational model (NO-SQL db). This type of db makes the tables independent from each other, not joined by foreign keys like SQL db. Then, executing a query is faster than searching all the tables separately and joining them with the JOIN clause like in SQL db. On the other hand, updates in NO-SQL db are not as efficient as in SQL ones because the programmer has to update all the related tables.

G) Example in postresql:

```
SELECT p.pers_name
FROM student s JOIN person p ON (s.student_id = p.person_id)
WHERE ( EXTRACT(year from NOW()) - EXTRACT(year from p.person_birth_date) )
BETWEEN (19 AND 21);
```

To optimize this query i would implement and index ordered by person_birth_date field. Making a binary search it will discard the persons with age less than 19 or more than 21.

H) With a stored procedure passing the request data as parameter. When calling the procedure we pass a student for example as an array of primitive type values and there is an INSERT implementation inside the procedure to save the student values in the student table.

An advantage of using a stored procedure would be to have all the business logic in one place. Also, it can be used from more than one application.
On the other hand, disadvantages include more difficult to resolve problems not only related to data and needs the programmers have great knowledge about sql language.