

En este taller utilizaremos PostgreSQL, un DBMS de código abierto. Éste se encuentra instalado en Linux en la sala Ada Lovelace y tiene creado un usuario *lovelace* y una base de datos por defecto del mismo nombre. También utilizamos pgAdmin4, una aplicación web para administrar bases de datos en PostgreSQL.

1. Accediendo a la BD desde un aplicación

1. Utilice el archivo `Taller3_base.sql` para crear una base de datos `taller3` en PostgreSQL.
2. Ejecute el archivo `EjemploConexionPostgreSQL_taller3_v2.sql` conectándose a la BD `taller3`. Asegúrese de entender todos los comandos que allí se encuentran.
3. Utilice el archivo `Taller4_base1.sql` para crear una base de datos `taller4a` en PostgreSQL.
4. Cree una consulta parametrizada en Java para obtener todos los cursos y notas dado un código de estudiante ingresado por el usuario. Use la clase `Statement`. Para solicitar el input del usuario puede por ejemplo usar la clase `Scanner`.
5. Ataque su código anterior con una inyección que retorne los cursos y notas de TODOS los estudiantes.
6. Corrija su código usando la clase `PreparedStatement`

2. SQL procedimental

1. Utilice el archivo `Taller4_base2.sql` para crear una base de datos `taller4b` en PostgreSQL.
2. Sobre la BD `taller4b` ejecute todas las funciones de las diapositivas 4-SQLAvanzado.
3. Escriba una función que retorne la razón de número de grupos sobre número de profesores.
4. Escriba una función que retorne la cadena de caracteres «Razón inferior a 2» si la razón del numeral anterior es menor a 2. De lo contrario retorna «Razón igual o superior a 2».
5. Re-escriba la función del numeral anterior (4) para que use internamente la función del numeral 3.
6. Cree una función que retorne una table con el nombre de cada unidad y el número de cursos ofrecidos por cada unidad.