

Acceso a Datos – Tarea 04

Desarrollo de aplicaciones multiplataforma

Felipe Rodriguez

Contenido

[Ejercicio 1 2](#_Toc510021781)

[Creando la clase director: 2](#_Toc510021782)

[Creando la clase película 2](#_Toc510021783)

[Creando los objetos directores y objetos 4](#_Toc510021784)

[Almacenando las películas 4](#_Toc510021785)

[Método Main inicial de pruebas 4](#_Toc510021786)

[Comprobando que funciona (1) 5](#_Toc510021787)

[Comprobando que funciona (2) 6](#_Toc510021788)

[Ejercicio 2 7](#_Toc510021789)

[Introduciendo la contraseña 7](#_Toc510021790)

[Borrando todos los datos 7](#_Toc510021791)

[Creando el tipo vivienda, dirección y la función publicidad 8](#_Toc510021792)

[Creando los tipos vivienda y dirección 8](#_Toc510021793)

[Creando la función publicidad 8](#_Toc510021794)

[Creando la tabla viviendas\_alquiler y sus consultas 8](#_Toc510021795)

[Creando la tabla viviendas\_alquiler 8](#_Toc510021796)

[Insertando registros de ejemplo 9](#_Toc510021797)

[Consulta de todas las ciudades diferentes existentes en la BD 9](#_Toc510021798)

[Consulta de todas las viviendas dadas por una ciudad 9](#_Toc510021799)

[Devolviendo el valor publicidad de una vivienda dada 9](#_Toc510021800)

[Probando el funcionamiento de todo 10](#_Toc510021801)

[Consultando la primera vivienda inicial 10](#_Toc510021802)

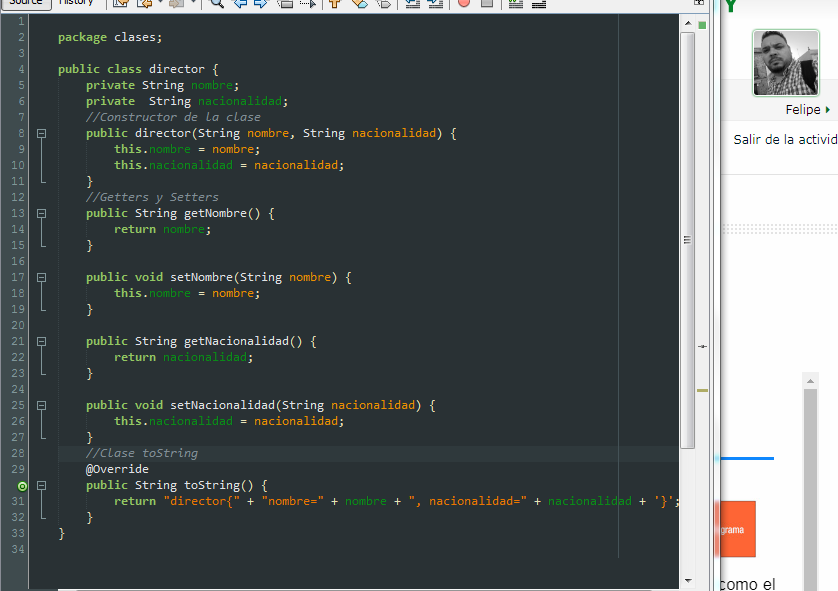
[Probando otra ciudad y una vivienda aleatoria 10](#_Toc510021803)

[Ejercicio 3 11](#_Toc510021804)

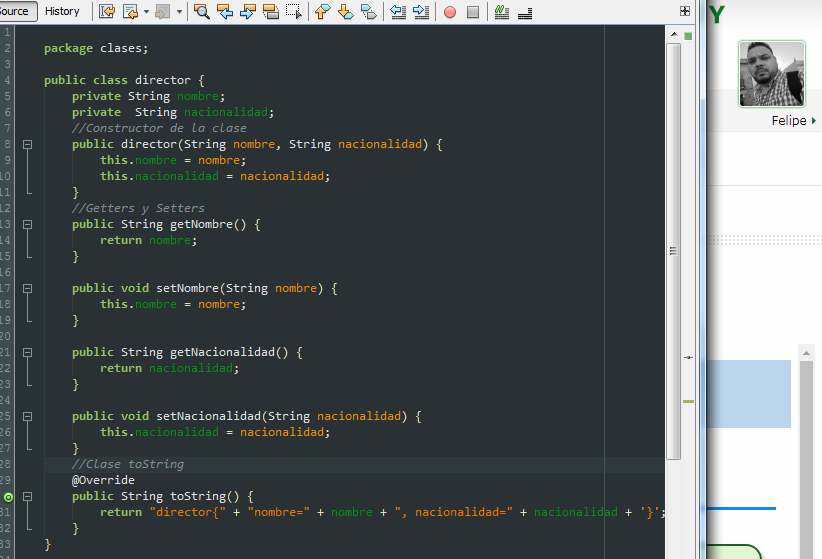
# Ejercicio 1

*NOTA: Obviaremos la parte de la descarga e instalación de la librería Db4o en nuestro proyecto al considerar que ya se encuentra documentado en el temario. Asimismo, no se han realizado tareas de validación de los datos de entrada (se considera que los datos que se van a introducir son válidos), dado que se considera que no son la finalidad de esta tarea y sopesaría más carga de trabajo que la tarea en si misma.*

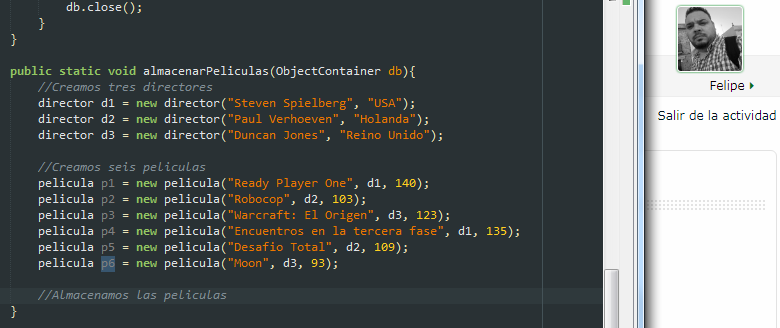
## Creando la clase director:



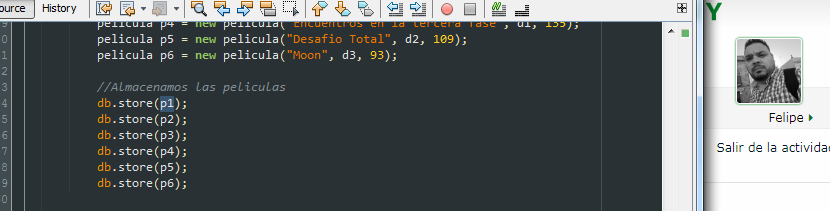
## Creando la clase película



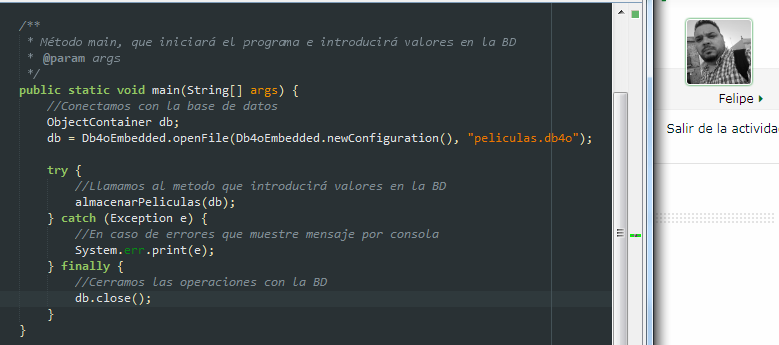
## Creando los objetos directores y objetos



## Almacenando las películas

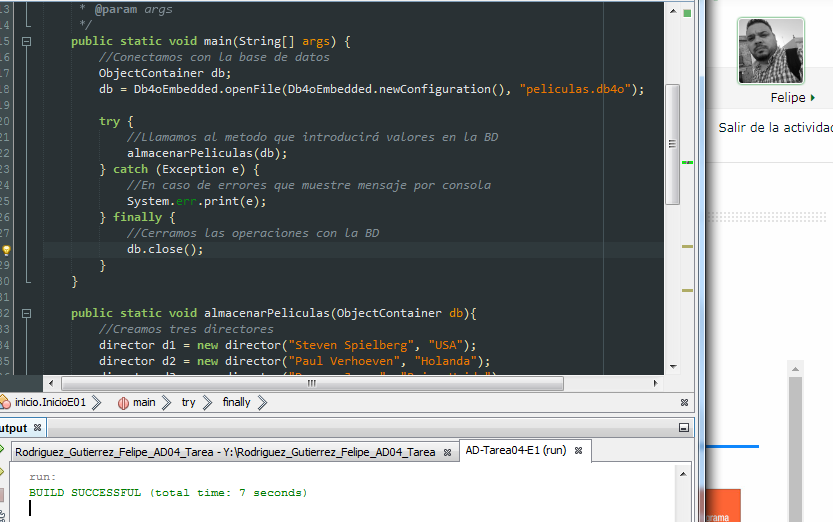


## Método Main inicial de pruebas

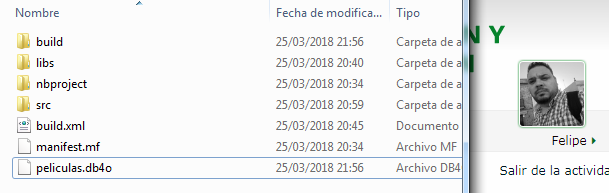


## Comprobando que funciona (1)

Con este método inicial, si ejecutamos el mismo, podremos observar por consola que crea satisfactoriamente la base de datos.

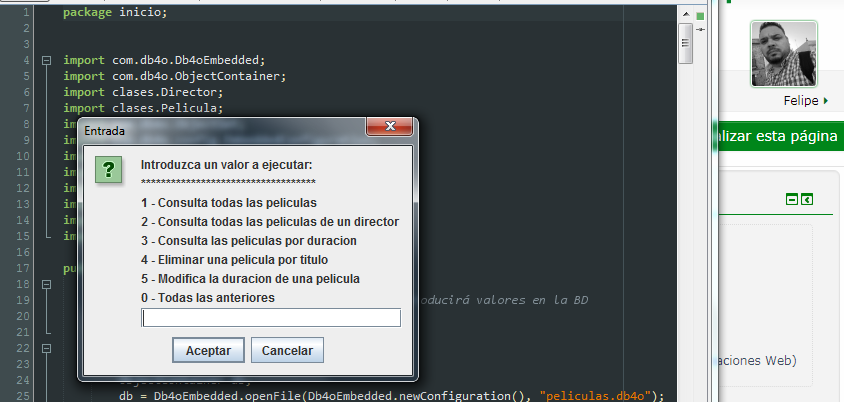


Si miramos en el directorio de nuestro proyecto, podremos observar que se ha creado la misma en su interior como se muestra en la imagen siguiente.

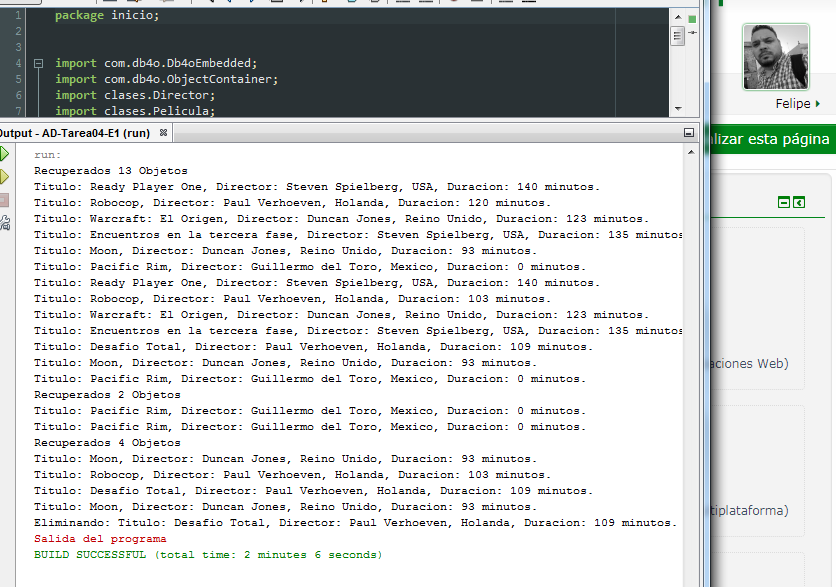


Ahora, añadiremos el resto de funcionalidades hasta estar completa, las cuales al ejecutarse las mismas, mostraran los resultados requeridos en pantalla.

## Comprobando que funciona (2)



Ejecutando el punto “Todas las anteriores”, mostrará por consola todas las acciones de golpe. Los resultados mostrados pueden ser diferentes a los que se puedan ver en otras ejecuciones (el ejercicio se distribuirá sin el archivo películas.db4o eliminado para que se inicie desde cero).

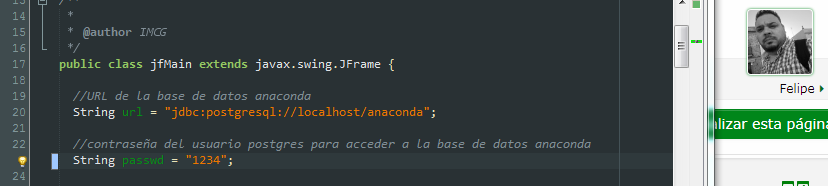


# Ejercicio 2

*NOTA: Obviaremos la parte de la descarga e instalación de PostgreSQL tanto en el sistema como en nuestro proyecto al considerar que ya se encuentra documentado en el temario. Asimismo, no se han realizado tareas de validación de los datos de entrada (se considera que los datos que se van a introducir son válidos), dado que se considera que no son la finalidad de esta tarea y sopesaría más carga de trabajo que la tarea en si misma.*

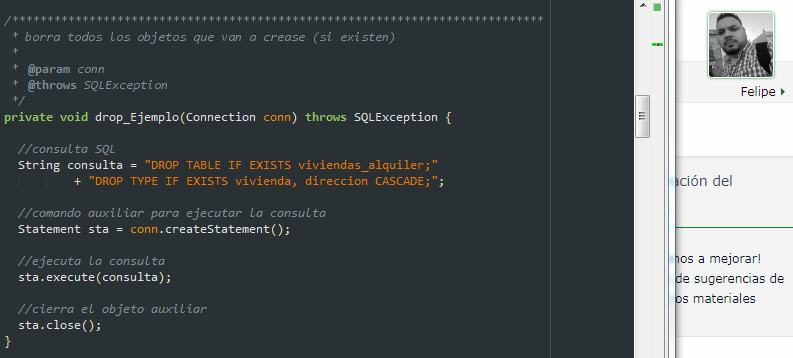
## Introduciendo la contraseña

Introduciendo la contraseña de PostgreSQL, que se ha definido en durante la instalación. En nuestro caso se ha utilizado la última versión disponible del gestor, 10.3. En nuestro caso, la contraseña que vamos a usar ya estaba definida en el proyecto.



## Borrando todos los datos

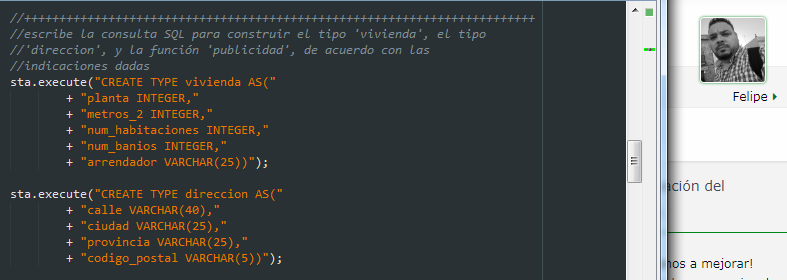
Este método viene predefinido con el proyecto de serie. No se le realizan cambios.



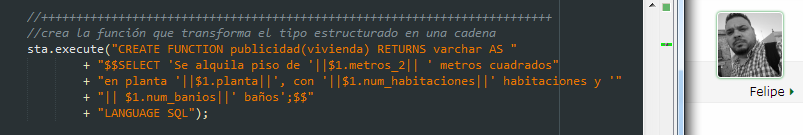
## Creando el tipo vivienda, dirección y la función publicidad

*NOTA: Se han modificado algunas líneas de orden y se han suprimido otras con motivo de aligerar líneas de código que se consideran innecesarias.*

### Creando los tipos vivienda y dirección



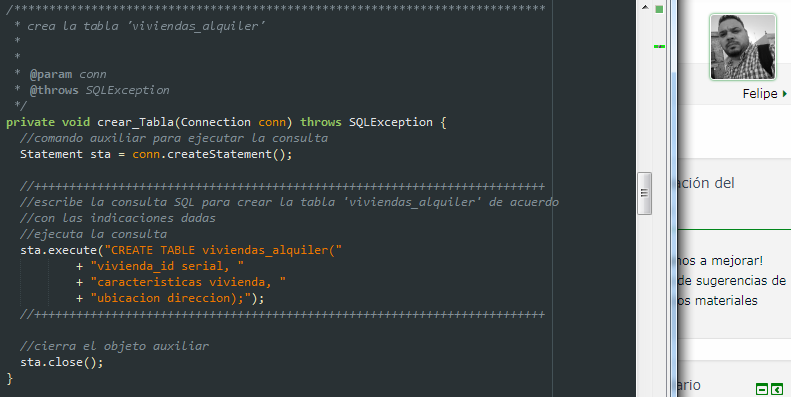
### Creando la función publicidad



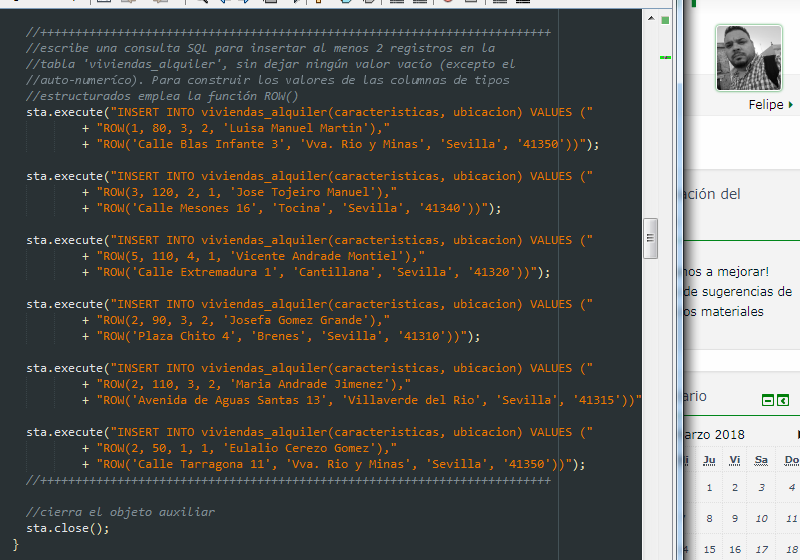
## Creando la tabla viviendas\_alquiler y sus consultas

*NOTA: Se han modificado algunas líneas de orden y se han suprimido otras con motivo de aligerar líneas de código que se consideran innecesarias.*

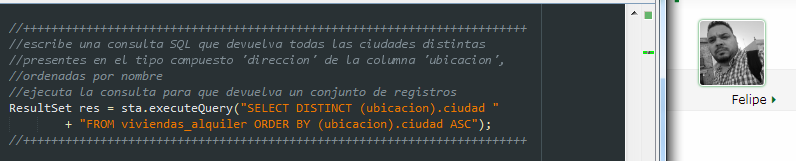
### Creando la tabla viviendas\_alquiler



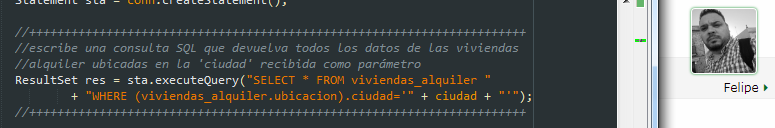
### Insertando registros de ejemplo



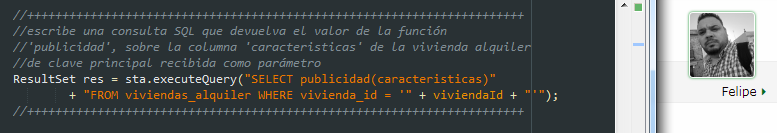
### Consulta de todas las ciudades diferentes existentes en la BD



### Consulta de todas las viviendas dadas por una ciudad



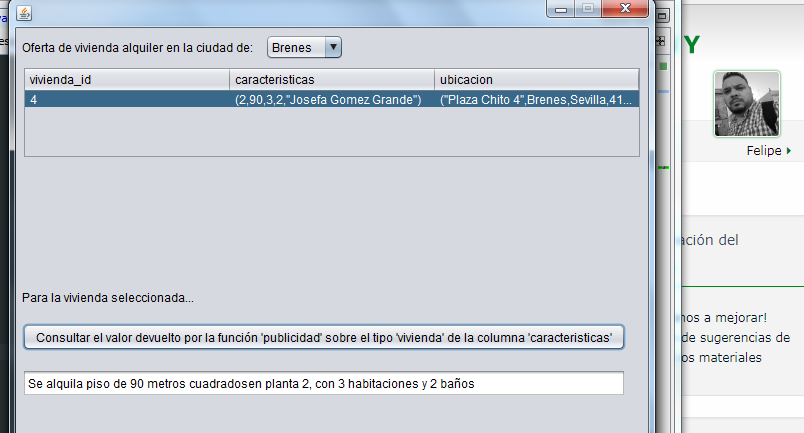
### Devolviendo el valor publicidad de una vivienda dada



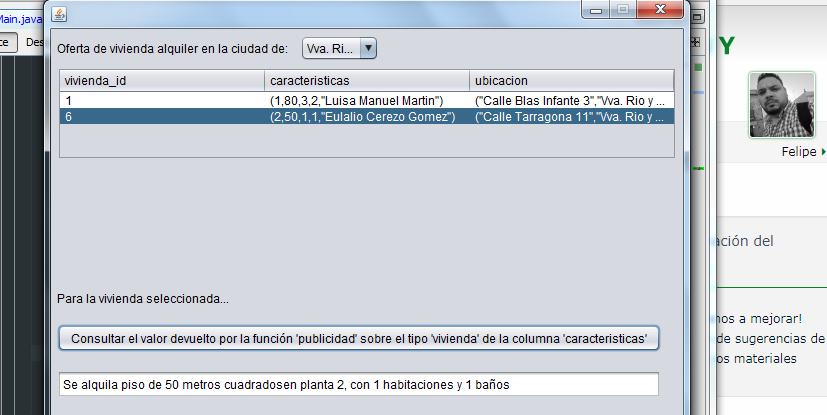
## Probando el funcionamiento de todo

*NOTA: Se le ha añadido una línea al código para centrar el cuadro en la ventana del sistema.*

### Consultando la primera vivienda inicial



### Probando otra ciudad y una vivienda aleatoria



# Ejercicio 3

## Conclusiones personales

Como conclusiones personales la verdad es que me ha gustado esta tarea, me ha resultado muy accesible con los ejemplos a descargar. En esta ocasión ha sido más intuitiva, considero que es muy muy similar a SQL, cosa que facilita mucho la tarea. Trabajar con objetos en esta ocasión me ha parecido mucho más  intuitivo y sencillo.

### Mejoras en mi tarea:

Como mejoras en mi programa, principalmente haría en la primera tarea una verificación de los datos de entrada, pero nos desviaríamos de la línea de esta tarea y de este módulo. Tal vez aplicar una interfaz gráfica en el primer ejercicio.

### Dificultades que te has encontrado:

Sin querer parecer presuntuoso, no me ha supuesto gran dificultad siguiendo los ejemplos y el temario (consideré Hibernate más enrevesado). Tal vez la consulta de las ciudades se atragantó el tiempo de irme a hacer una café.

### Opinión personal del trabajo realizado:

Añadir a mi opinión personal, pues me ha gustado trabajar con db4o por poder incorporar una base de datos funcional en su propia aplicación con solo adjuntar la librería, aunque su “deprecación” por no existir una línea continuista en esta aplicación, hace que alguien que quiera desarrollar una aplicación quiera usar estas bases de datos que, pese a que funcionan, no tienen evolución. Con PostgreSQL se conoce que hay mercado laboral y es muy solicitado en ofertas de empleo. La única “pega” que le encuentro es que nuestra aplicación depende de un servidor de base de datos externo, o al menos es lo que llevo visto con el curso como en esta tarea.

