

Construir una clase **Habitacion** que tenga tres atributos: **numero** (de tipo **int**), **metros** (de tipo **double**), y **baño** (de tipo **boolean**). El constructor de esta clase tendrá tres parámetros y asignará valores a estos atributos. Además, habrá tres métodos getters, uno por atributo. La clase no tendrá setters; una vez asignados valores en el constructor no se podrán modificar de ninguna forma (lo que se conoce como una "clase inmutable").

La idea es que esta clase representará una habitación de un hotel, donde el **numero** representa el número de habitación, los **metros** representan los metros cuadrados de la habitación y **baño** indica si la habitación tiene baño o no.

Construir también una clase **Hotel**, que tendrá **Habitaciones** (objetos de la clase creada anteriormente), y donde cada **Habitación** puede estar reservada o no.

El constructor de esta clase creará un **Hotel** vacío (sin habitaciones). En esta clase, añadir los siguientes métodos:

boolean añadir (Habitacion h)

Añade una habitación al hotel, siempre que en el hotel no haya ya una habitación con el mismo **numero**. Devolverá **true** si la habitación se añade correctamente y **false** si no. Tras añadir la habitación, ésta no estará reservada.

boolean borrar (int numero)

Borra del hotel la habitación con el **numero** indicado. Devuelve **true** si la habitación se borra correctamente, y **false** si no (porque no exista una habitación con ese **numero**).

double totalMetros()

Devuelve el número total de **metros** cuadrados de todas las habitaciones del hotel juntas.

boolean reservar (int numero)

Reserva la habitación que tenga el número indicado. Devolverá **true** si la habitación se reserva correctamente, y **false** si no (ya sea porque la habitación ya estaba reservada anteriormente, o porque no existe una habitación con ese **numero**).

boolean anular (int numero)

Anula la reserva de la habitación que tenga el **numero** indicado. Devolverá **true** si la reserva se anula correctamente, y **false** si no (ya sea porque la habitación no estaba reservada anteriormente, o porque no existe una habitación con ese **numero**).

int buscar1 (double metrosMinimo, boolean baño)

Busca una habitación libre (no reservada) que cumpla los requisitos indicados, es decir, que tenga como mínimo el número de **metros** indicado, y que tenga baño o no según indique el parámetro **baño**. El funcionamiento del parámetro **baño** es como sigue: si el parámetro **baño** es **true**, sólo aceptamos habitaciones con baño. Si el parámetro **baño** es **false**, aceptamos habitaciones tanto con como sin baño. El valor devuelto por este método será un **numero** de habitación disponible, o **-1** si no hay ninguna que cumpla los requisitos.

int [] buscarN (double metrosMinimo, boolean baño, int cuantas)

Busca **cuantas** habitaciones libres (no reservadas), que cumplan los requisitos indicados, y que estén todas en la misma planta. Devuelve un array de **int** con **cuantas** elementos, con los **numeros** de habitación encontrados. Si no hay un conjunto de habitaciones libres en el hotel que cumpla los requisitos, devolverá **null**.

La planta en que se encuentra una habitación es la cifra de centenas del **numero** de habitación: la habitación 104 está en la planta 1, la habitación 511 está en la planta 5, la habitación 201 está en la planta 2, etc.