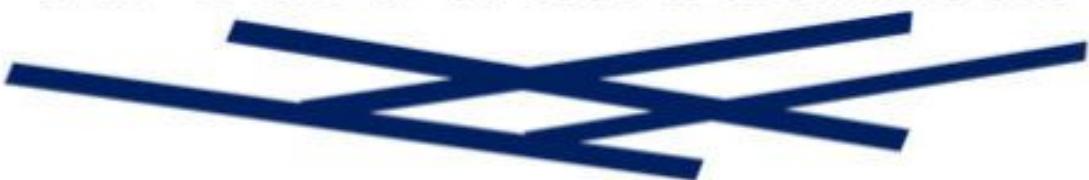




# Tu Casa Inmobiliaria



Proyecto desarrollado por: Daniel Ivaylov Gerchev

Desarrollo de aplicaciones web

IES la vereda

**I.E.S. La Vereda**

*La Pobla de Vallbona - València*



# Índice

1. Introducción a la aplicación	página 3
2. Manual de usuario:	
2.1 Aplicación de navegación libre:	página 5
2.1.1 Barra de navegación y portada	
2.1.2 Sección de inmuebles	
2.1.3 Buscar inmuebles	
2.1.4 Filtro de datos.	
2.1.5 Resultado de la búsqueda	
2.1.6 Inmueble en detalle.	
2.1.7 Contacto e información adicional.	
2.2 Aplicación de cara al propietario de inmuebles.	página 10
2.2.1 Registro	
2.2.1 Loguear usuario	
2.2.2 Función añadir inmueble	
2.2.3 Función borrar inmueble	
2.3 Aplicación de cara empleado.	página 13
2.3.1 Solicitar credenciales y entrar.	
2.3.2 Funciones de empleado.	
2.4 Aplicación de cara al administrador	página 15
3 Manual técnico:	página 15
3.1 Máquina virtual	
3.2 Servidor local Xampp	página 16
3.3 MySQL	
3.3.1 Planteamiento de bases de datos.	
3.3.2 Creación de bases de datos en remoto.	
3.3.3 Configuración de archivo Connection de php.	
3.3.4 Consultas a la base de datos	
3.4 Programación	página 27
3.4.1 Lógica de la programación y clases	
3.4.2 Estructuras de control y objetos	
3.4.3 Lectura, escritura de información, almacenamiento.	
3.4.4 Sesiones y seguridad de usuarios.	
3.4.5 Conexión a base de datos en local	
3.5 Diseño de interfaces y lenguaje de marcas.	página 32
3.5.1 Elección de estilos, colores, fuentes, planteamiento de diseño.	
3.5.2 Lenguaje de marcas	
3.5.2.1 HTML	
3.5.2.2 CSS	
3.5.2.3 Bootstrap	
3.5.2.4 Gimp	
3.6 Entornos de desarrollo.	página 36

3.6.1 Metodología de desarrollo del software	
3.6.2 Sistema de gestión de versiones	
3.6.3 Plan de pruebas. Caja blanca	
3.7 Despliegue de aplicaciones	página 39
3.8 Entorno cliente.	página 42
4 Problemas	página 44
5 Mejorar la aplicación	página 44
6 El código	página 44-126

## 1 Introducción a la aplicación:

---

### **El precio de la vivienda sube al mayor ritmo en una década**

---

**Madrid y Cataluña registran los incrementos más altos, con el 10% y el 9**

<https://www.lavanguardia.com/economia/20190308/46913344722/vivienda-precio-in-dice-nueva-usada.html>

Es tan solo una de muchas noticias que nos “bombardean” no solo en periódicos sino que sale todos los meses incluso en la tele como “índicador” económico de que la crisis se va y vuelve la fiebre por comprar vivienda. Dios nos guarde pero... Si no puedes con ellos... únete! Pero añadiría: innova, busca llamar más clientes que la competencia y no hagas las cosas a la antigua usanza... automatiza tareas repetitivas y hunde la competencia con una plataforma mejor.

Porque el mercado del ladrillo es una buena apuesta si tras años de crisis la gente no aprendió que precisamente una de las causas fueron la burbuja inmobiliaria?

Por una sencilla razón. Toda persona necesita vivir en su propia casa, así que donde hay un bien de necesidad básica hay un negocio grande detrás.



El precio de las viviendas sube de nuevo. Es inevitable que una persona quiera comprar o alquilar una casa. Después de investigar el funcionamiento de las inmobiliarias descubrí que todas comparten una misma política a la hora de captar y “atar” al propietario que da su confianza a la inmobiliaria para alquilar o vender inmueble. Todas las inmobiliarias te piden que firmes por contrato una exclusividad durante muchos meses con ellos...

Que pasa si la inmobiliaria está saturada? Que pasa si no están interesados en vender tu inmueble por sus bajas comisiones y prefieren vender casas de lujo que tienen una comisión más jugosa? Que pasa si el mismo propietario decide que ya no quiere vender/alquilar su inmueble? Al final tu inmueble pasa a coger polvo y tu tienes una permanencia con ellos durante meses y tienes que esperar todo ese tiempo sin que puedas hacer “nada” ya que si se lo das a otra inmobiliaria o lo vendes por tu cuenta tendrás que pagarles la comisión igualmente.

Aquí es donde surge tu casa inmobiliaria. Hemos encontrado una necesidad de los clientes la cual es ser “Libres” en cuanto a disponer con su inmueble. Si tu inmueble no se vende lo puedes dar de baja tu mismo. El propietario sube a nuestra plataforma el inmueble que desea por lo tanto agilizamos el trabajo de nuestros empleados ya que no tendrán que llenar formularios de inmuebles nuevos y el usuario estará feliz porque sabe lo que ha hecho y sabe que en cualquier momento puede disponer de lo que es suyo sin dar dinero.

## 2 Manual de usuario:

---

A continuación el manual de usuario el cual nos será muy útil a la hora de instruir los empleados y clientes. La aplicación se basa en tres tipos de usuarios que tienen un rol diferente y específico. Mas adelante se podrán añadir mas funciones según el rol o añadir funciones compartidas para reutilizar código pero de esto se hablará más adelante

## 2.1 Aplicación de navegación libre:

Habrá una parte en la que no será necesario ingresar con una cuenta ya que nos interesa que los clientes que buscan inmuebles para comprar/alquilar no encuentren dificultades añadidas y buscar inmueble sea lo más fácil.

Inicio de la aplicación:

Nos muestra la marca, y el valor añadido de la empresa.

### 2.1.1 Barra de navegación y portada

Una sencilla barra de navegación que tendrá:

- 1 **Inicio** que lleva a la portada de la página.
- 2 **Buscar** inmuebles que a su vez dentro tendrá sección de alquiler o venta.
- 3 **Registrar** usuario. Se podrá registrar para poder gestionar sus inmuebles.
- 4 **Contacto**. Más información sobre la empresa y cómo pueden contactar con nosotros.
- 5 **Login**. Los usuarios registrados podrán entrar y hacer las funciones específicas de cada uno.



### 2.1.2 Sección de inmuebles

En la parte posterior habrá una sección con los inmuebles destacados de la inmobiliaria con información más importante del inmueble. Foto, tipo, habitaciones, metros y altura. El

inmueble tiene un número de referencia con el cual usted podrá pedir mas información o solicitar una cita para ver el inmueble al personal cualificado de nuestra inmobiliaria.

## A continuacion nuestra mejor seleccion de inmuebles

			
<b>Ciudad aasdf</b> Informacion sobre inmueble: Tipo de inmuble: chalet Habitaciones: 1 Metros cuadrados: 1111 Altura: 1 Numero de referencia: <input type="text" value="1"/>	<b>Ciudad Valencia</b> Informacion sobre inmueble: Tipo de inmuble: chalet Habitaciones: 4 Metros cuadrados: 120 Altura: 1 Numero de referencia: <input type="text" value="3"/>	<b>Ciudad Valencia</b> Informacion sobre inmueble: Tipo de inmuble: piso Habitaciones: 4 Metros cuadrados: 120 Altura: 1 Numero de referencia: <input type="text" value="4"/>	<b>Ciudad Valencia</b> Informacion sobre inmueble: Tipo de inmuble: chalet Habitaciones: 1 Metros cuadrados: 120 Altura: 1 Numero de referencia: <input type="text" value="5"/>
<a href="#">Mas informacion</a>	<a href="#">Mas informacion</a>	<a href="#">Mas informacion</a>	<a href="#">Mas informacion</a>

Notas internas: Siguiendo la psicología humana, una imagen vale más que mil palabras. No se indicará si el piso esta en venta o alquiler ya que una persona puede encontrar algo que encaja en su búsqueda de alquiler pero al darle y ver que está en venta podría llegar a plantearse su compra.

### 2.1.3 Buscar inmuebles

En la segunda pestaña tendra una parte común a todos que sera buscar inmuebles. Se puede seleccionar alquilar o comprar inmueble.



The screenshot shows the main navigation bar with links for 'Tu casa', 'Inicio', 'Buscar inmuebles', 'Registro usuarios', 'Contacto', and 'Login'. Below the navigation is a logo featuring a house icon above the text 'Tu Casa Inmobiliaria' with a stylized 'X' underneath. Two large buttons are displayed: 'Alquilar' (Rental) and 'Comprar' (Purchase), each with a star rating below it. At the bottom, there's a footer with links for 'Daniel Ivaylov', 'Inicio', 'Inmobiliaria en: C/San Sebastian 5 Valencia CP 46019', 'Inform', and a small 'Inmobiliaria' logo.

El cual nos redirigirá a la sección de filtrar nuestra búsqueda.

## 2.1.4 Filtro de datos.

Se podrá filtrar por los factores más importantes a la hora de buscar una vivienda.

The screenshot shows a search interface with four main filters:

- Dinero:** An image of a large blue Euro symbol on a pile of money. Below it is a text input field labeled "Cuanto es tu presupuesto maximo?" (What is your maximum budget?).
- Tipo de vivienda:** An image of a city skyline. Below it is a dropdown menu currently set to "chalet".
- Ciudad:** An image of a modern building complex at night. Below it is a dropdown menu currently set to "aasdf".
- Habitaciones:** An image of a house floor plan. Below it is a dropdown menu currently set to "Sin habitaciones".

At the bottom center is a blue "Buscar" (Search) button. At the very bottom, there is a footer bar with the text "Daniel Ivaylov", "Inicio", "Inmobiliaria en: C/San Sebastian 5 - 16 161 Valencia CP 46019".

Si no se introduce bien el dato se le indicará o si no se encuentra lo que busca le redirigirá al mismo sitio.



## 2.1.5 Resultado de la búsqueda

Al enviar su búsqueda aparecerán los inmuebles que encajan en las características y una flecha grande para volver atrás y volver a hacer otra búsqueda si es necesario.

Nota interna: Ahora si imprimimos el precio del alquiler del piso para que la gente evalúe según precio ya que es una búsqueda concreta de venta o alquiler



Ciudad Valencia

Informacion sobre inmueble:

Tipo de inmueble: piso  
Alquiler: 450  
Metros cuadrados: 120  
Habitaciones: 4  
Altura: 1  
Número de referencia:

[Mas informacion](#)



Ciudad Valencia

Informacion sobre inmueble:

Tipo de inmueble: piso  
Alquiler: 800  
Metros cuadrados: 140  
Habitaciones: 4  
Altura: 1  
Número de referencia:

[Mas informacion](#)



VOLVER ATRAS

← →

## 2.1.6 Inmueble en detalle.

Si le damos a más información entraremos a ver el piso en detalle. Podremos ver casi toda la información de interés de cara al cliente. Metros cuadrados, ascensor comentarios , todas las imágenes, extendidos etc.

Tu casaInicioBuscar inmueblesRegistro usuariosContactoLogin ▾



Referencia:	13	Tipo:	chalet
Alquiler:	333	Venta:	333
Habitaciones:	1	Metros:	33
Ciudad:	aasdf	Provincia:	alava
Calle:	asdf	Numero:	33
CP:	3333	Ciudad:	aasdf
Altura:	1	Ascensor:	No
Comentario:	asdf		

## 2.1.7 Contacto e información adicional.



### Inmobiliaria tu casa

El **valor añadido** de nuestra empresa consiste en que no existe la palabra permanencia. Usted no se atara a través de un contrato de exclusividad con nosotros como hacen todas las demás inmobiliarias. Confiamos tanto en la **profesionalidad de nuestros agentes** que estamos seguros de que su inmueble se venderá/alquilará rápido y si no es así usted mismo puede dar de baja el inmueble que ha registrado en nuestra página web.

### Donde nos encontramos.

Ciudad:	Valencia	CP:	46019
Calle:	C/San Sebastian 5	Telefono:	961 616 161
Email:	tucasa@ieslavereda.es	Responsable	Daniel Ivaylov Gerchev

Le ofrecemos asesoramiento, visitas guiadas por nuestros agentes a clientes y tasación totalmente gratuitos. No le cobraremos ni un céntimo hasta consigamos vender/alquilar su inmueble. No tendrá que lidiar con usuarios molestos ni personas curiosas.

### Registrar usuario

Si se registra como usuario podrá gestionar todos sus inmuebles, para alquilar o vender online, de manera muy sencilla. Quiere dar de alta un inmueble En tan solo 30 segundos ya estará subido online y la gente puede recibir toda la información sobre él.

[Registrarse](#)

### Buscar inmuebles

Puede buscar inmuebles para comprar y alquilar por toda España. Tan fácil como filtrar por el dinero máximo que desea gastarse, la ciudad en la que desea vivir, el número de habitaciones y el tipo de inmueble.

[Buscar](#)

## 2.2 Aplicación de cara al propietario de inmuebles.

Para que un propietario pueda tener cuenta y suba sus inmuebles necesita registrarse en la pestaña de registrar usuario. Para que se registre una persona hace falta que se rellenen todos los campos. En cada campo hay un ejemplo que indica de qué forma se puede introducir el dato, además se iluminará en verde si es correcto o en rojo si tiene que editarse. Si el campo está vacío y se da a enviar indicará los campos que faltan por llenar.

### 2.2.1 Registro

*Regístrate para anunciar tus inmuebles totalmente gratis!*

Usuario

Nombre

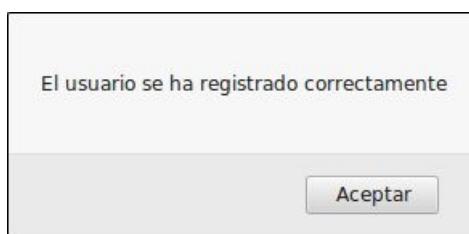
Apellido

Teléfono

Email  
  
Nunca vamos a dar su información a terceros.

Contraseña

Si el usuario se ha registrado correctamente lo indicará



## 2.2.1 Loguear usuario

Tu casa Inicio Buscar inmuebles Registro usuarios Contacto Login ▾

*Si te has registrado puedes loguear para gestionar tus inmuebles*

Usuario  
Ejemplo: TzarDaniel

Contraseña  
Introduce contraseña

Enviar

Daniel Naylov Inicio Inmobiliaria en: C/San Sebastián 5 Valencia CP 46019 Información al 961 816 161

Una vez registrado te logueas y puedes hacer las funciones de un usuario. Tanto si no se ha introducido nombre de usuario, contraseña o alguno de los datos esta mal introducido se le indicará tras hacer una consulta a la base de datos.



## 2.2.2 Función añadir inmueble

Si se ha logueado correctamente se le dará la bienvenida indicando su nombre y los datos que usted nos ha facilitado para que nos pongamos en contacto en caso de que haya interesados en sus inmuebles.

*Aquí puedes añadir y borrar tus pisos, dani*

Si alguien está interesado en sus inmuebles nosotros nos pondremos en contacto con usted con los datos que nos ha facilitado, estos no serán públicos y siempre que alguien le llame será un profesional de nuestra inmobiliaria:

Telefono:

666666666

Correo:

paco@go.com

Para registrar un piso necesita llenar el formulario

## Registrar nuevo inmueble

Precio alquiler <input type="text" value="Entero"/>	Precio venta <input type="text" value="Entero"/>	Metros cuadrados <input type="text" value="Metros"/>	Calle <input type="text" value="Nombre de la calle"/>
Numero <input type="text" value="Entero"/>	Puerta <input type="text" value="Entero"/>	Ciudad <input type="text" value="First name"/>	Telefono <input type="text" value="Entero"/>
Tipo inmueble <input type="text" value="Chalet"/>		Ascensor <input type="text" value="No"/>	Habitaciones <input type="text" value="1"/>
Piso <input type="text" value="1"/>	Provincia <input type="text" value="Alava"/>	Codigo postal <input type="text" value="CP"/>	
Foto 1 <input type="button" value="Examinar..."/>	Foto 2 <input type="button" value="Examinar..."/>	Foto 3 <input type="button" value="Examinar..."/>	No se ha seleccionado ningún archivo.
Foto 4 <input type="button" value="Examinar..."/>	Foto 5 <input type="button" value="Examinar..."/>	No se ha seleccionado ningún archivo.	
Descripcion del inmueble <input type="text" value="Describenos de la mejor manera las características del inmueble. Recuerda que de eso depende su venta/alquiler."/>			

Todos los campos tienen que estar rellenos y en verde como en el caso anterior. Al darle en el botón de foto 1,2 etc aparecerá una ventana en la que podrás buscar la imagen para subir a nuestra base de datos.

## Registrar nuevo inmueble

Precio alquiler <input type="text" value="1000"/>	Precio venta <input type="text" value="diasdf"/>	Metros cuadrados <input type="text" value="33"/>	Calle <input type="text" value="calle123"/>																																							
Numero <input type="text" value="123"/>	Puerta <input type="text" value="123"/>	Ciudad <input type="text" value="Barakaldo"/>	Telefono <input type="text" value="0000000000"/>																																							
<div style="border: 1px solid #ccc; padding: 5px;"> <p>Carga de archivos</p> <table border="1"> <thead> <tr> <th>Nombre</th> <th>Tamaño</th> <th>Modificado</th> </tr> </thead> <tbody> <tr><td>winmarkId-BootstrapFormHelpers...</td><td>27 abr</td><td></td></tr> <tr><td>tablon</td><td>7ene</td><td></td></tr> <tr><td>startbootstrap-small-business-gh-p...</td><td>jue</td><td></td></tr> <tr><td>startbootstrap-heroic-features-gh-p...</td><td>27 abr</td><td></td></tr> <tr><td>ProyectoFinalInmobiliaria</td><td>18 abr</td><td></td></tr> <tr><td>PHPveregalos</td><td>22 ene</td><td></td></tr> <tr><td>JsonYAjax</td><td>20 feb</td><td></td></tr> <tr><td>Inmobiliaria Tu Casa_files</td><td>18 abr</td><td></td></tr> <tr><td>inmobiliaria</td><td>Ayer</td><td></td></tr> <tr><td>deporDanie.com11042019</td><td>11 abr</td><td></td></tr> <tr><td>CopiasSeguridadInmobiliaria</td><td>Ayer</td><td></td></tr> <tr><td>copiaSeguridadPHP</td><td>18 feb</td><td></td></tr> </tbody> </table>  </div>				Nombre	Tamaño	Modificado	winmarkId-BootstrapFormHelpers...	27 abr		tablon	7ene		startbootstrap-small-business-gh-p...	jue		startbootstrap-heroic-features-gh-p...	27 abr		ProyectoFinalInmobiliaria	18 abr		PHPveregalos	22 ene		JsonYAjax	20 feb		Inmobiliaria Tu Casa_files	18 abr		inmobiliaria	Ayer		deporDanie.com11042019	11 abr		CopiasSeguridadInmobiliaria	Ayer		copiaSeguridadPHP	18 feb	
Nombre	Tamaño	Modificado																																								
winmarkId-BootstrapFormHelpers...	27 abr																																									
tablon	7ene																																									
startbootstrap-small-business-gh-p...	jue																																									
startbootstrap-heroic-features-gh-p...	27 abr																																									
ProyectoFinalInmobiliaria	18 abr																																									
PHPveregalos	22 ene																																									
JsonYAjax	20 feb																																									
Inmobiliaria Tu Casa_files	18 abr																																									
inmobiliaria	Ayer																																									
deporDanie.com11042019	11 abr																																									
CopiasSeguridadInmobiliaria	Ayer																																									
copiaSeguridadPHP	18 feb																																									
<div style="border: 1px solid #ccc; padding: 5px;"> <p>Habitaciones <input type="text" value="1"/></p> <p>Codigo postal <input type="text" value="33333"/></p> <p>Foto 3 <input type="button" value="Examinar..."/> No se ha seleccionado ningún archivo.</p> </div>																																										

### 2.2.3 Función borrar inmueble

Si pinchas encima de esta imagen podrás borrar el inmueble que necesitas



Nota interna: Está hecho al propósito que esto sea diferente y se oculte a la vista con una selección de colores de letras claras que no llamen la atención y se difuminan con el resto de la página. No nos interesa que el usuario borre sus anuncios y está al final de todo.

Al seleccionar esto nos dirigirá en una pagina donde disponemos de todos nuestros inmuebles, indicando nuestro nombre e id de usuario para asegurarnos que es nuestro. Al darle borrar se borrará para siempre de la base de datos.

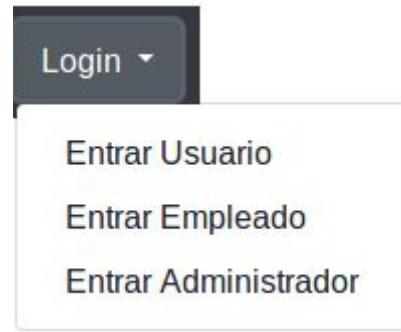
*Estos son todos tus inmuebles, dani. Si quieres borrar alguno solo tienes que seleccionarlo. Si deseas volver atrás puedes hacerlo al final de la pagina.*

<p>Ciudad Valencia Informacion sobre inmueble:  Id propietario: 1 Tipo de inmueble: piso Habitaciones: 4 Metros cuadrados: 120 Altura: 1 Numero de referencia:  4  Borrar inmueble</p>	<p>Ciudad Valencia Informacion sobre inmueble:  Id propietario: 1 Tipo de inmueble: chalet Habitaciones: 1 Metros cuadrados: 120 Altura: 1 Numero de referencia:  5  Borrar inmueble</p>	<p>Ciudad Valencia Informacion sobre inmueble:  Id propietario: 1 Tipo de inmueble: piso Habitaciones: 4 Metros cuadrados: 140 Altura: 1 Numero de referencia:  8  Borrar inmueble</p>	<p>Ciudad Valencia Informacion sobre inmueble:  Id propietario: 1 Tipo de inmueble: chalet Habitaciones: 1 Metros cuadrados: 33 Altura: 1 Numero de referencia:  9  Borrar inmueble</p>
--	--	--	---

## 2.3 Aplicación de cara empleado.

### 2.3.1 Solicitar credenciales y entrar.

Un empleado podrá solicitar su usuario y contraseña al administrador. Al obtenerlas se puede loguear en la parte derecha de la aplicación.



Tu casa Inicio Buscar inmuebles Registro usuarios Contacto Login ▾

**Sección de empleados. Ingresa tus datos para vender y alquilar inmuebles.**

Usuario  
Ejemplo: TzarDaniel

Contraseña  
Introduce contraseña

Enviar

### 2.3.2 Funciones de empleado.

El empleado podrá buscar información, alquilar, vender o borrar inmueble por el número de referencia.

**Podeis obtener informacion de los inmuebles, alquilar, vender, y borrar pisos.**

Si deseas obtener mas datos para informar a los interesados sobre el inmueble busca por **numero de referencia**

4

Enviar

Selecciona piso para alquilar por su **numero de referencia**

4

Enviar

Selecciona piso para vender de la base de datos por su **numero de referencia**

5

Enviar

Selecciona piso para borrar de la base de datos por su **numero de referencia**

4

Enviar

## 2.4 Aplicación de cara al administrador

The screenshot shows a web application interface for administrator registration. At the top, there is a navigation bar with links: 'Tu casa', 'Inicio', 'Buscar inmuebles', 'Registro usuarios', 'Contacto', and a 'Login' button. Below the navigation bar, there is a heading in Spanish: 'Sección de administrador, introduce tus datos a continuación para dar de alta nuevos empleados.' The form fields are as follows:

- Usuario:** A text input field with placeholder text 'Ejemplo: TzarDaniel'.
- Contraseña:** A text input field with placeholder text 'Introduce contraseña'.
- Enviar:** A grey 'Enviar' (Send) button.
- Registration Information:** A section with the text 'Registra a tus empleados para que puedan vender, alquilar y borrar pisos.' followed by several input fields:
  - Nombre:** A text input field with placeholder text 'Ejemplo: TzarDaniel'.
  - Apellido:** A text input field with placeholder text 'Ejemplo: Daniel'.
  - Teléfono:** A text input field with placeholder text 'Ejemplo: 653366987'.
  - Email:** A text input field with placeholder text 'Ejemplo: divaylov@ieslavereda.es' and a small note below it: 'Nevera vamos a dar su información a terceros.'
  - Contraseña:** A text input field with placeholder text 'Introduce contraseña'.
- Enviar:** A grey 'Enviar' (Send) button.

## 3 Manual técnico:

---

Para el proyecto se han utilizado las siguientes tecnologías:

html, css ,bootstrap,gimp , php, consultas mysql, phpmyadmin,xamp, máquina virtual, los editores de texto php storm y sublime text, git, 000webhost, javascript

Se verá una muestra en general de todas a continuación.

### 3.1 Máquina virtual



Se ha creado una máquina virtual para trabajar en local, poder llevarnos el trabajo a donde sea con nosotros en un disco duro con todos los editores de texto y programas necesarios para que el servidor que tenemos instalado en local funcione y podamos crear copias de seguridad de la misma máquina virtual. En mi caso tengo una copia de seguridad de la máquina que actualizo cada semana.

### 3.2 Servidor local Xampp

XAMPP es un paquete de software libre, que consiste principalmente en el sistema de gestión de bases de datos MySQL, el servidor web Apache y los intérpretes para lenguajes de script PHP y Perl. El nombre es en realidad un acrónimo: X, Apache, MariaDB/MySQL, PHP, Perl.

A screenshot of the XAMPP website. The header is dark blue with white text. It includes links for 'Apache Friends', 'Applications', 'FAQs', 'HOW-TO Guides', 'PHPInfo', and 'phpMyAdmin'. Below the header, there is a large orange button with the XAMPP logo and the text 'XAMPP Apache + MariaDB + PHP + Perl'. The main content area is light gray with a thin horizontal line separating the header from the body.

### Welcome to XAMPP for Linux 7.2.9

Instalación servidor local.

Descargamos de <https://www.apachefriends.org/es/index.html>

Le damos permisos

```
sudo chmod +x xampp-linux-x64-7.2.9-0-installer.run
```

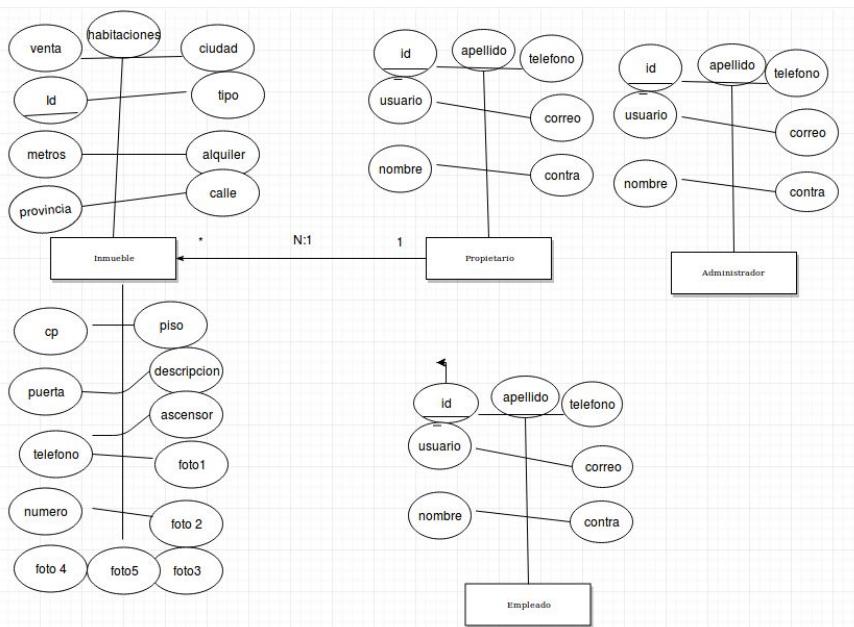
Ejecutamos archivo

```
sudo ./xampp-linux-x64-7.2.9-0-installer.run
```

Para levantar el servidor

## 3.3 MySQL

### 3.3.1 Planteamiento de bases de datos.



Con el programa de diagram.draw.io se ha hecho el diagrama entidad relación que luego se ha generado en mysql y coinciden.

Tabla	Acción	Filas	Tipo	Cotejamiento	Tamaño	Residuo a depurar
administrador	Examinar Estructura Buscar Insertar Vaciar Eliminar	1	InnoDB	utf8_spanish_ci	48 KB	-
empleado	Examinar Estructura Buscar Insertar Vaciar Eliminar	4	InnoDB	utf8_spanish_ci	48 KB	-
inmueble	Examinar Estructura Buscar Insertar Vaciar Eliminar	11	InnoDB	utf8_spanish_ci	32 KB	-
propietario	Examinar Estructura Buscar Insertar Vaciar Eliminar	14	InnoDB	utf8_spanish_ci	48 KB	-
4 tablas	Número de filas	30	InnoDB	latin1_swedish_ci	176 KB	0 B

Tabla **inmueble**

id- sera identificador del inmueble por el cual se podrá buscar, referencia

tipo- será por ejemplo piso, chalet etc

alquiler- necesitamos la columna de alquiler y venta porque el mismo inmueble se podrá vender o alquilar a la vez.

venta

habitaciones

metros

precio

provincia

ciudad  
 calle  
 cp-código postal  
 número-de la calle  
 puerta  
 piso- la altura de la vivienda  
 ascensor- será si o no  
 descripción- se añadirá la descripción del inmueble  
 teléfono- sera el teléfono del propietario que verán solo los empleados

idPropietario- se obtiene de la sesión iniciada. necesario para ver los pisos que tiene un propietario que podrá tener más de uno obviamente. Será clave foránea para hacer la relación entre propietario e inmuebles que puede tener este propietario. Puede tener más de uno

Acción	Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo	Comentario
<a href="#">Editar</a> <a href="#">Eliminar</a>	<b>PRIMARY</b>	BTREE	Si	No	id	9	A	No	
<a href="#">Editar</a> <a href="#">Eliminar</a>	<b>fk_inmueble_propietario</b>	BTREE	No	No	idPropietario	2	A	No	
#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra	
<input type="checkbox"/>	1 <b>id</b>	int(4)			No	Ninguna		AUTO_INCREMENT	
<input type="checkbox"/>	2 <b>tipo</b>	varchar(255)	utf8_spanish_ci		No	Ninguna			
<input type="checkbox"/>	3 <b>alquiler</b>	int(255)			Sí	NULL			
<input type="checkbox"/>	4 <b>venta</b>	int(255)			Sí	NULL			
<input type="checkbox"/>	5 <b>habitaciones</b>	int(10)			No	Ninguna			
<input type="checkbox"/>	6 <b>metros</b>	int(255)			No	Ninguna			
<input type="checkbox"/>	7 <b>provincia</b>	varchar(255)	utf8_spanish_ci		No	Ninguna			
<input type="checkbox"/>	8 <b>ciudad</b>	varchar(255)	utf8_spanish_ci		No	Ninguna			
<input type="checkbox"/>	9 <b>calle</b>	varchar(255)	utf8_spanish_ci		No	Ninguna			
<input type="checkbox"/>	10 <b>cp</b>	int(10)			No	Ninguna			
<input type="checkbox"/>	11 <b>numero</b>	int(5)			No	Ninguna			
<input type="checkbox"/>	12 <b>puerta</b>	int(5)			No	Ninguna			
<input type="checkbox"/>	13 <b>piso</b>	int(5)			No	Ninguna			
<input type="checkbox"/>	14 <b>idPropietario</b>	int(255)			No	Ninguna			
<input type="checkbox"/>	15 <b>telefonoPropietario</b> int(10)				No	Ninguna			
<input type="checkbox"/>	16 <b>descripcion</b>	varchar(255)	utf8_spanish_ci		No	Ninguna			
<input type="checkbox"/>	17 <b>ascensor</b>	varchar(22)	utf8_spanish_ci		No	Ninguna			
<input type="checkbox"/>	18 <b>foto1</b>	varchar(50)	utf8_spanish_ci		No	Ninguna			
<input type="checkbox"/>	19 <b>foto2</b>	varchar(50)	utf8_spanish_ci		Sí	NULL			
<input type="checkbox"/>	20 <b>foto3</b>	varchar(50)	utf8_spanish_ci		Sí	NULL			
<input type="checkbox"/>	21 <b>foto4</b>	varchar(50)	utf8_spanish_ci		Sí	NULL			
<input type="checkbox"/>	22 <b>foto5</b>	varchar(50)	utf8_spanish_ci		Sí	NULL			

### Tabla **propietario**

id  
 usuario-único  
 nombre  
 apellido  
 teléfono

correo  
contra-contraseña

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra
1	<b>id</b>	int(255)			No	Ninguna		AUTO_INCREMENT
2	<b>usuario</b>	varchar(255)	utf8_spanish_ci		No	Ninguna		
3	<b>nombre</b>	varchar(255)	utf8_spanish_ci		No	Ninguna		
4	<b>telefono</b>	int(255)			No	Ninguna		
5	<b>correo</b>	varchar(255)	utf8_spanish_ci		No	Ninguna		
6	<b>contra</b>	varchar(255)	utf8_spanish_ci		No	Ninguna		
7	<b>apellido</b>	varchar(255)	utf8_spanish_ci		No	Ninguna		

↑  Seleccionar todo Para los elementos que están marcados:  Cambiar Eliminar Primero

Imprimir Planteamiento de la estructura de tabla Mover columnas Normalizar

Agregar 1 columna(s) después de apellido Continuar

### Índices

Acción	Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo
Editar  Eliminar	<b>PRIMARY</b>	BTREE	Sí	No	id	14	A	No
Editar  Eliminar	<b>usuario</b>	BTREE	Sí	No	usuario	14	A	No
Editar  Eliminar	<b>correo</b>	BTREE	Sí	No	correo	14	A	No

## Tabla empleado

id  
usuario-único  
nombre  
apellido  
teléfono  
correo-único  
contra-contraseña

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra
1	<b>id</b>	int(255)			No	Ninguna		AUTO_INCREMENT
2	<b>usuario</b>	varchar(255)	utf8_spanish_ci		No	Ninguna		
3	<b>nombre</b>	varchar(255)	utf8_spanish_ci		No	Ninguna		
4	<b>telefono</b>	int(255)			No	Ninguna		
5	<b>correo</b>	varchar(255)	utf8_spanish_ci		No	Ninguna		
6	<b>contra</b>	varchar(255)	utf8_spanish_ci		No	Ninguna		
7	<b>apellido</b>	varchar(255)	utf8_spanish_ci		No	Ninguna		

↑  Seleccionar todo Para los elementos que están marcados:  Cambiar Eliminar Primero

Imprimir Planteamiento de la estructura de tabla Mover columnas Normalizar

Agregar 1 columna(s) después de apellido Continuar

### Índices

Acción	Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo
Editar  Eliminar	<b>PRIMARY</b>	BTREE	Sí	No	id	2	A	No
Editar  Eliminar	<b>usuario</b>	BTREE	Sí	No	usuario	2	A	No
Editar  Eliminar	<b>correo</b>	BTREE	Sí	No	correo	2	A	No

## Tabla administrador

id

usuario-único

nombre

apellido

teléfono

correo

contra-contraseña

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra	Acciones
1	<b>id</b>	int(255)			No	Ninguna		AUTO_INCREMENT	
2	<b>usuario</b>	varchar(255)	utf8_spanish_ci		No	Ninguna			
3	<b>nombre</b>	varchar(255)	utf8_spanish_ci		No	Ninguna			
4	<b>telefono</b>	int(255)			No	Ninguna			
5	<b>correo</b>	varchar(255)	utf8_spanish_ci		No	Ninguna			
6	<b>contra</b>	varchar(255)	utf8_spanish_ci		No	Ninguna			
7	<b>apellido</b>	varchar(255)	utf8_spanish_ci		No	Ninguna			

↑  Seleccionar todo Para los elementos que están marcados:  Examinar Cambiar Eliminar Primario

Imprimir Planteamiento de la estructura de tabla Mover columnas Normalizar

Agregar 1 columna(s) después de apellido

**índices**

Acción	Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo
Editar  Eliminar	<b>PRIMARY</b>	BTREE	Sí	No	id	0	A	No
Editar  Eliminar	<b>usuario</b>	BTREE	Sí	No	usuario	0	A	No
Editar  Eliminar	<b>correo</b>	BTREE	Sí	No	correo	0	A	No

v id3315951_inmobiliaria2 administrador	
	id : int(255)
	usuario : varchar(255)
	nombre : varchar(255)
	telefono : int(255)
	correo : varchar(255)
	contra : varchar(255)
	apellido : varchar(255)

v id3315951_inmobiliaria2 inmueble	
	id : int(4)
	tipo : varchar(255)
	alquiler : int(255)
	venta : int(255)
	habitaciones : int(10)
	metros : int(255)
	provincia : varchar(255)
	ciudad : varchar(255)
	calle : varchar(255)
	cp : int(10)
	numero : int(5)
	puerta : int(5)
	piso : int(5)
	idPropietario : int(255)
	telefonoPropietario : int(10)
	descripcion : varchar(255)
	ascensor : varchar(22)
	foto1 : varchar(50)
	foto2 : varchar(50)
	foto3 : varchar(50)
	foto4 : varchar(50)
	foto5 : varchar(50)

v id3315951_inmobiliaria2 propietario	
	id : int(255)
	usuario : varchar(255)
	nombre : varchar(255)
	telefono : int(255)
	correo : varchar(255)
	contra : varchar(255)
	apellido : varchar(255)

v id3315951_inmobiliaria2 empleado	
	id : int(255)
	usuario : varchar(255)
	nombre : varchar(255)
	telefono : int(255)
	correo : varchar(255)
	contra : varchar(255)
	apellido : varchar(255)

### 3.3.2 Creación de bases de datos en remoto.

Aparte del despliegue en local después de crear la base de datos se ha generado un archivo SQL para subirlo al servidor remoto de <https://www.000webhost.com/>

The screenshot shows the 000webhost dashboard. At the top, there are several menu items: Build Website, Website Manager, Learn to Code, Power Store, Settings, and Community. A red button labeled "SPECIAL DEALS" is visible. Below the menu, a survey asks, "How likely are you to recommend 000webhost to a friend ?" with a scale from 0 (Never recommend) to 10 (Extremely likely). The "Used Databases" section shows one database named ".000webhostapp.com" with an "ADD MORE RESOURCE" button. A table titled "manage databases" lists one database entry:

DB Name	DB User	DB Host
id3315951_inmobiliaria2	id3315951_inmobiliaria2	localhost

Por desgracia no fue posible importar el archivo ya que dio error y se han tenido que ejecutar todos los script en la pestaña de sql. Todo fue bien de esa forma y la base de datos en remoto está operativa.

## Importando al servidor actual

### Archivo a importar:

El archivo puede ser comprimido (gzip, bzip2, zip) o descomprimido.  
A compressed file's name must end in **[format].[compression]**. Example: **.sql.zip**

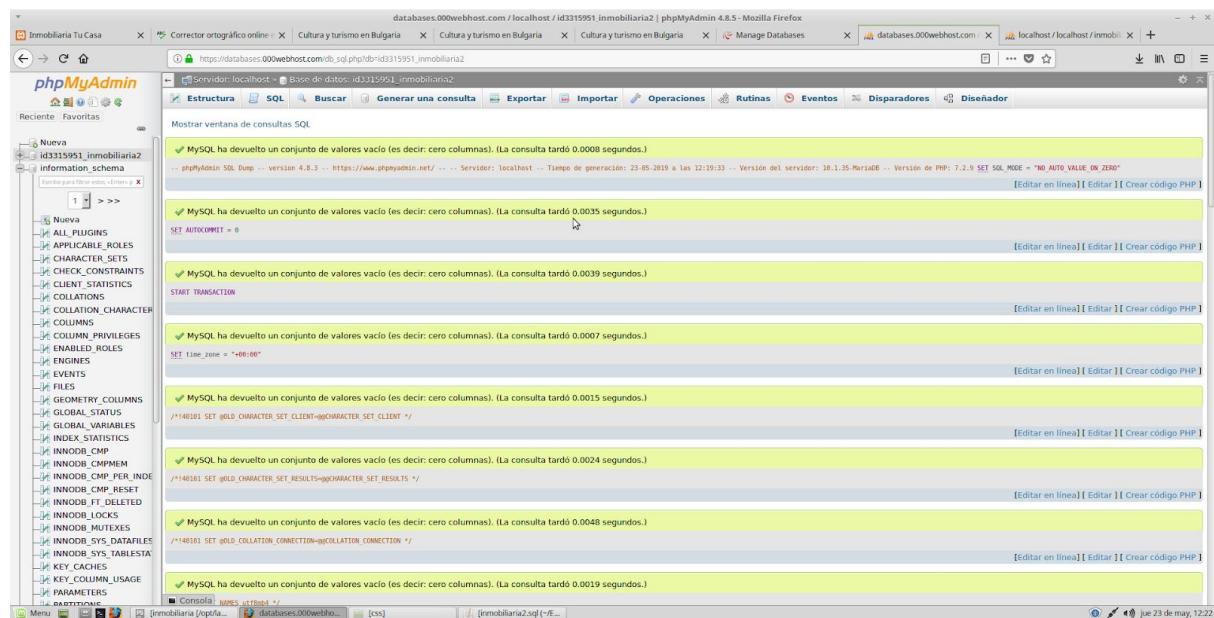
Buscar en su ordenador:  Examinar... No se ha seleccionado ningún archivo. (Máximo: 16MB) También puede arrastrar un archivo en cualquier página.

Por lo tanto se procedió a ejecutar los script de la base de datos manualmente. en la pestaña de SQL

Las sentencias DDL se utilizan para crear y modificar la estructura de las tablas así como otros objetos de la base de datos.

- CREATE - para crear objetos en la base de datos.
- ALTER - modifica la estructura de la base de datos.
- DROP - borra objetos de la base de datos.
- TRUNCATE - elimina todos los registros de la tabla, incluyendo todos los espacios asignados a los registros.

Podemos ver algunas de las sentencias a continuación.



The screenshot shows the phpMyAdmin interface with the SQL history panel open. The panel displays several executed MySQL queries, each with a green checkmark indicating success. The queries are as follows:

```
MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0.00008 segundos.)  
-- phpMyAdmin SQL Dump -- version 4.8.3 -- https://www.phpmyadmin.net/ -- Server: localhost -- Tiempo de generación: 23-05-2019 a las 12:39:33 -- Versión del servidor: 10.1.35-MariaDB -- Versión de PHP: 7.2.9 SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO"  
[Editar en línea] [Editar] [Crear código PHP]  
  
MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0.0035 segundos.)  
SET AUTOCOMMIT = 0  
[Editar en línea] [Editar] [Crear código PHP]  
  
MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0.0039 segundos.)  
START TRANSACTION  
[Editar en línea] [Editar] [Crear código PHP]  
  
MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0.0007 segundos.)  
SET time_zone = "+00:00"  
[Editar en línea] [Editar] [Crear código PHP]  
  
MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0.0015 segundos.)  
/*140181 SET @OLD_CHARACTER_SET_CLIENT=@CHARACTER_SET_CLIENT */  
[Editar en línea] [Editar] [Crear código PHP]  
  
MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0.0024 segundos.)  
/*140181 SET @OLD_CHARACTER_SET_RESULTS=@CHARACTER_SET_RESULTS */  
[Editar en línea] [Editar] [Crear código PHP]  
  
MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0.0048 segundos.)  
/*140181 SET @OLD_COLLATION_CONNECTION=@COLLATION_CONNECTION */  
[Editar en línea] [Editar] [Crear código PHP]  
  
MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0.0019 segundos.)
```

### 3.3.3 Configuración de archivo Connection de php.

Para que podamos acceder a la base de datos se ha de crear una clase Connection en php en la que introduciremos los siguientes datos.

## /public\_html/Connection.php

```
1 <?php
2 /**
3  * Created by PhpStorm.
4  * User: dani
5  * Date: 7/01/19
6  * Time: 16:39
7 */
8
9 class Connection
10 {
11     public static function make(){
12         try{
13             $opciones=[PDO::MYSQL_ATTR_INIT_COMMAND=>"SET NAMES utf8",
14                         PDO::ATTR_ERRMODE=>PDO::ERRMODE_EXCEPTION,
15                         PDO::ATTR_PERSISTENT=>true];
16             $conection=new PDO(
17                 'mysql:localhost;dbname=id3315951_inmobiliaria2;charset=utf8',
18                 'id3315951_inmobiliaria2',
19                 '',
20                 $opciones);
21
22         }catch (PDOException $PDOException){
23             die($PDOException->getMessage());
24         }
25     return $conection;
26 }
```

Los datos nos los proporciona el webhost además podemos personalizarlo poniendo en este caso inmobiliaria dos. En la linea donde esta el cursor ha de introducirse la contraseña que en este caso está borrada por razones de seguridad.

DB Name	DB User	DB Host	
id3315951_inmobili...	id3315951_inmobili...	localhost	Manage ▾

### 3.3.4 Consultas a la base de datos

Lenguaje de manipulación de datos (DML)

Las sentencias de lenguaje de manipulación de datos (DML) son utilizadas para gestionar datos dentro de los schemas. Algunos ejemplos:

- SELECT - para obtener datos de una base de datos.
- INSERT - para insertar datos a una tabla.
- UPDATE - para modificar datos existentes dentro de una tabla.

A la hora de acceder a una página web se ejecuta automáticamente el método get para obtener información de la base de datos. En este caso obtendremos objetos de la clase inmueble sin repetir para poder insertar todas las ciudades tan solo 1 vez gracias al comando group by ciudad/tipo.

Eso nos servira para facilitar la búsqueda de pisos por ciudad sin que se repita y así evitamos que el usuario meta manualmente nombre de ciudades y estas se puedan repetir.

```
><?php
ob_start();

require __DIR__."/partials/inmobiliaria.part.php";
require_once 'Connection.php';
$PDO=Connection::make();

if ($_SERVER['REQUEST_METHOD']=='GET') {
    //obtenemos todos los inmuebles agrupando por ciudad para que no nos salga mas de una vez l
    $stmt=$PDO->prepare( statement: "SELECT * FROM inmueble group by ciudad");
    $stmt->setFetchMode( mode: PDO::FETCH_CLASS|PDO::FETCH_PROPS_LATE, classNameObject: "inmueble"
    $stmt->execute();
    $ciudades=$stmt->fetchAll();

    //obtenemos todos los inmuebles por tipo y agrupando para que no se repita
    $stmt=$PDO->prepare( statement: "SELECT * FROM inmueble group by tipo");
    $stmt->setFetchMode( mode: PDO::FETCH_CLASS|PDO::FETCH_PROPS_LATE, classNameObject: "inmueble"
    $stmt->execute();
    $tipos=$stmt->fetchAll();
}

}
```

Un problema con el cual me he encontrado ha sido que se insertaba un espacio y a la hora de imprimir los objetos me daba error .

Un **Lenguaje de Control de Datos (DCL)** por sus siglas en inglés: **Data Control Language**) es un lenguaje proporcionado por el [Sistema de Gestión de Base de Datos](#) que incluye una serie de comandos SQL que permiten al administrador controlar el acceso a los datos contenidos en la [Base de Datos](#).

- CONNECT
- SELECT
- INSERT
- UPDATE
- DELETE
- USAGE

En mi proyecto se han usado múltiples select, insert , delete. Unos ejemplos son por ejemplo el filtrado que hago por búsqueda de alquileres y compras.

```
$statement=$PDO->prepare("SELECT * FROM inmueble where alquiler<:dinero and alquiler>0 and tipo = :tipo and ciudad = :ciudad and habitaciones>= :habitaciones");
```

En este ejemplo pasamos el presupuesto máximo que tiene la persona como parámetro dinero para que salgan inmuebles que se ajustan al bolsillo del comprador y que sea mayor que 0 ya que si es 0 significa que el inmueble no se vende/alquila. Después filtramos por el tipo de inmueble, ciudad seleccionada y habitaciones. En el caso de habitaciones normalmente la gente busca un mínimo de habitaciones por lo tanto las habitaciones tiene que ser mayor o igual al número de habitaciones que le pasamos.

```
ob_start();
require __DIR__."/partials/inmobiliaria.part.php";
require_once "Connection.php";
$PDO=Connection::make();
$objetos=[];

if ($_SERVER['REQUEST_METHOD']=='POST') {
//SELECT * FROM inmueble where alquiler< 5000 and tipo = 'chalet' and ciudad = 'Valencia' and habitaciones< 4
$dinero=$_POST['dinero'];
$tipo=$_POST['tipo'];
$ciudad=$_POST['ciudad'];
$habitaciones=$_POST['habitaciones'];

$stmt=$PDO->prepare( "SELECT * FROM inmueble where alquiler<:dinero and alquiler>0 and tipo = :tipo and ciudad = :ciudad and habitaciones>= :habitaciones");
$stmt->setFetchMode( mode: PDO::FETCH_CLASS|PDO::FETCH_PROPS_LATE, classNameObject: "inmueble");
$stmt->bindParam( parameter: ':dinero', &variable: $dinero);
$stmt->bindParam( parameter: ':tipo', &variable: $tipo);
$stmt->bindParam( parameter: ':ciudad', &variable: $ciudad);
$stmt->bindParam( parameter: ':habitaciones', &variable: $habitaciones);
$stmt->execute();
$objetos=$stmt->fetchAll();
// print_r($objetos);

if (!empty($objetos)) {
//no hay objetos redirigimos atras
header( string: 'Location:alquilar.php');
}
}
```

El insert más destacado sería el de crear un piso por la cantidad de datos y el control de las mismas en el cual controlamos las variables obligatorias y las que pueden estar vacías como por ejemplo el precio de alquilar o comprar piso.

```

$sql="INSERT INTO inmueble (tipo , alquiler, venta, habitaciones, metros,provincia, ciudad, calle, cp, numero, puerta, p
$tipo=$_POST['tipo'];
$habitaciones=$_POST['habitaciones'];
$metros=$_POST['metros'];
$provincia=$_POST['provincia'];
$ciudad=$_POST['ciudad'];
$calle=$_POST['calle'];
$cp=$_POST['cp'];
$direccion=$_POST['direccion'];
$piso=$_POST['piso'];
$telefonoPropietario=$_POST['telefonoPropietario'];
$descripcion=$_POST['descripcion'];
$ascensor=$_POST['ascensor'];

if(!isset($salquiler)){
    $salquiler=$_POST['alquiler'];
}
if(!isset($venta)){
    $venta=$_POST['venta'];
}

```

Controlamos que se pueda insertar un piso con mínimo una foto y opcional meter hasta 5. Se pueden añadir las fotos que uno desea pero he puesto límite 5 para que no haya muchas fotos en los archivos.

```

if (is_file($foto1) === true) {
    $name1 = $idUnico . '_f1' . $name1;
}if (move_uploaded_file($_FILES['foto1']['tmp_name'], $foto1)) {

    $statement = $PDO->prepare($sql);
    $statement->bindParam( parameter: ':tipo', &variable: $tipo);
    $statement->bindParam( parameter: ':habitaciones', &variable: $habitaciones);
    $statement->bindParam( parameter: ':metros', &variable: $metros);
    $statement->bindParam( parameter: ':provincia', &variable: $provincia);
    $statement->bindParam( parameter: ':ciudad', &variable: $ciudad);
    $statement->bindParam( parameter: ':calle', &variable: $calle);
    $statement->bindParam( parameter: ':telefonoPropietario', &variable: $telefonoPropietario);
    $statement->bindParam( parameter: ':cp', &variable: $cp);
    $statement->bindParam( parameter: ':numero', &variable: $numero);
    $statement->bindParam( parameter: ':puerta', &variable: $puerta);
    $statement->bindParam( parameter: ':piso', &variable: $piso);
    $statement->bindParam( parameter: ':descripcion', &variable: $descripcion);
    $statement->bindParam( parameter: ':ascensor', &variable: $ascensor);
    $statement->bindParam( parameter: ':foto1', &variable: $foto1);

    $statement->bindParam( parameter: ':idPropietario', &variable: $idPropietario);

    if (isset($venta)) {
        $statement->bindParam( parameter: ':venta', &variable: $venta);
    }
    if (isset($alquiler)) {
        $statement->bindParam( parameter: ':alquiler', &variable: $alquiler);
    }

    if (is_file($foto2) === true) {
        $name2 = $idUnico . '_f2' . $name2;
        if (move_uploaded_file($_FILES['foto2']['tmp_name'], $foto2)) {
            $statement->bindParam( parameter: ':foto2', &variable: $foto2);
        }
    }else{
        $foto2=null;
        $statement->bindParam( parameter: ':foto2', &variable: $foto2);
    }

    if (is_file($foto3) === true) {
        $name3 = $idUnico . '_f3' . $name3;
        if (move_uploaded_file($_FILES['foto3']['tmp_name'], $foto3)) {
            $statement->bindParam( parameter: ':foto3', &variable: $foto3);
        }
    }
}

```

## 3.4 Programación

### 3.4.1 Lógica de la programación y clases

Propietario, administrador y empleado tienen los mismos atributos por lo tanto a la hora de programar las clases para instanciar objetos se hará una clase genérica Persona y se implementaran las tres clase con la clase Persona. Al hacer esto nos ahorraremos muchísimo código ya que tendríamos que tener este archivo multiplicado por 3.

```

<?php
/* Created by PhpStorm. ...*/
class Persona
{
    protected $id;
    protected $usuario;
    protected $nombre;
    protected $telefono;
    protected $correo;
    protected $contra;
    protected $apellido;

    /**
     * Persona constructor.
     * @param $id
     * @param $usuario
     * @param $nombre
     * @param $telefono
     * @param $correo
     * @param $contra
     * @param $dni
     */
    public function __construct($id=null, $usuario=null, $nombre=null, $telefono=null, $correo=null, $contra=null, $apellido=null)
    {
        $this->id = $id;
        $this->usuario = $usuario;
        $this->nombre = $nombre;
        $this->telefono = $telefono;
        $this->correo = $correo;
        $this->contra = $contra;
        $this->apellido = $apellido;
    }

    /**
     * @return mixed
     */
    public function getId()
    {
        return $this->id;
    }
}

```

Y en cambio con decirle que propietario, administrador, empleado heredan de Persona nos ahorra muchísimo tiempo y esfuerzo.

```

<?php
/* Created by PhpStorm. ...*/

class Propietario extends Persona
{
}

```

### 3.4.2 Estructuras de control y objetos

En [lenguajes de programación](#), las **estructuras de control** permiten modificar el flujo de ejecución de las instrucciones de un [programa](#).

Con las estructuras de control se puede:

- De acuerdo con una condición, ejecutar un grupo u otro de sentencias (If-Then-Else)
- De acuerdo con el valor de una variable, ejecutar un grupo u otro de sentencias (Select-Case)
- Ejecutar un grupo de sentencias **mientras** se cumpla una condición (Do-While)
- Ejecutar un grupo de sentencias **hasta** que se cumpla una condición (Do-Until)

Se han usado muchas condiciones por ejemplo está en la que si tenemos una segunda foto la subira a la base de datos. Si no la hay estará a null y no nos dará problemas a la hora de insertarla.

```
if (is_file($foto2) === true) {
    $name2 = $idUnico . '_f2_' . $name2;
    if (move_uploaded_file($_FILES['foto2']['tmp_name'], $foto2)) {
        $statement->bindParam( parameter: ':foto2', &variable: $foto2);
    }
} else{
    $foto2=null;
    $statement->bindParam( parameter: ':foto2', &variable: $foto2);
}
```

Otro ejemplo interesante es la estructura de for each que imprimirá tantos objetos como encuentre de la clase Inmueble.

```
<h1 class="text-center ">A continuacion nuestra mejor seleccion de inmuebles</h1>
<div class="row text-center">
    <?php foreach($objetos as $objeto): ?>
        <div class="col-lg-3 col-md-6 mb-4">
            <div class="card h-100">
                <img class="card-img-top zoom" src=<?php echo $objeto['foto1'];?>>
                <div class="card-body">
                    <h4 class="card-title">Ciudad <?php echo $objeto['ciudad'];?> </h4>
                    <p class="card-text">
                        Informacion sobre inmueble:
                    </p>
                    <p class="text-left card-text">
                        Tipo de inmueble: <?php echo $objeto['tipo'];?><br>
                        Habitaciones: <?php echo $objeto['habitaciones'];?><br>
                        Metros cuadrados: <?php echo $objeto['metros'];?><br>
                        Altura: <?php echo $objeto['piso'];?><br>
                        Numero de referencia:
                    </p>
                    <form action="masInformacion.php" method="POST" enctype="multipart/form-data">
                        <input type="text" readonly name="id" value=<?php echo $objeto['id'];?> />
                    </form>
                <div class="card-footer">
                    <!-- class="btn btn-primary submit"-->
                    <input class="btn btn-primary" type="submit" value="Mas informacion"></input>
                </div>
            </div>
        </div>
    <?php endforeach; ?>
</div>
```

### 3.4.3 Lectura, escritura de información, almacenamiento.

Antes habíamos explicado el método get con el cual obtenemos información de la base de datos y la imprimimos.

```

<?php
ob_start();

require __DIR__."/partials/inmobiliaria.part.php";
require_once 'Connection.php';
$PDO=Connection::make();

if ($_SERVER['REQUEST_METHOD']=='GET') {
    //obtenemos todos los inmuebles agrupando por ciudad para que no nos salga mas de una vez las repetidas
    $stmt=$PDO->prepare( statement: "SELECT * FROM inmueble group by ciudad");
    $stmt->setFetchMode( mode: PDO::FETCH_CLASS|PDO::FETCH_PROPS_LATE, classNameObject: "inmueble");
    $stmt->execute();
    $ciudades=$stmt->fetchAll();

    //obtenemos todos los inmuebles por tipo y agrupando para que no se repita
    $stmt=$PDO->prepare( statement: "SELECT * FROM inmueble group by tipo");
    $stmt->setFetchMode( mode: PDO::FETCH_CLASS|PDO::FETCH_PROPS_LATE, classNameObject: "inmueble");
    $stmt->execute();
    $tipos=$stmt->fetchAll();
}

```

Con el método post enviamos información a la base de datos a través de un formulario

Formulario:

```

<form action="registrarUsuario.php" method="POST" enctype="multipart/form-data">
<h1>Registrate para anunciar tus inmuebles totalmente gratis!</h1>

<div class="form-group">
    <label class="text-left" for="formGroupExampleInput">Usuario</label>
    <input type="text" class="form-control" id="nombre" name="usuario" placeholder="Ejemplo: TzarDaniel" required>
</div>

<div class="form-group">
    <label class="text-left" for="formGroupExampleInput">Nombre</label>
    <input type="text" class="form-control" id="nombre" onblur="comprobarLetrasNombre();" name="nombre" placeholder="Ejemplo: Daniel" required>
</div>
<div class="form-group">
    <label class="text-left" for="formGroupExampleInput2">Apellido</label>
    <input type="text" class="form-control" id="apellido" onblur="comprobarLetrasApellido();" name="apellido" placeholder="Ejemplo: Gerchev" required>
</div>
<div class="form-group">
    <label class="text-left" for="formGroupExampleInput2">Telefono</label>
    <input type="text" class="form-control" id="telefono" onblur="comprobarTelefono();" name="telefono" placeholder="Ejemplo:653366987" required>
</div>

<div class="form-group">
    <label class="text-left" for="exampleInputEmail1">Email</label>
    <input type="email" class="form-control" aria-describedby="emailHelp" name="correo" placeholder="Ejemplo: divaylov@ieslavereda.es" required>
    <small class="form-text text-muted">Nunca vamos a dar su informacion a terceros.</small>
</div>
<div class="form-group">
    <label class="text-left" for="exampleInputPassword1">Contraseña</label>
    <input type="password" class="form-control" name="contra" placeholder="Introduce contraseña" required>
</div>

<input type="submit" value="Enviar">

```

Método post que comprobará si hay alguno de estos parámetros vacíos y devolverá una respuesta en caso de que hay alguno vacío y sino se ejecutara el statement e enviare la informacion a la base de datos

```

if ($_SERVER['REQUEST_METHOD']=='POST') {
    $sql="INSERT INTO propietario (usuario, nombre, apellido, telefono, correo,contra) VALUES (:usuario,:nombre,:apellido, :telefono, :correo, :contra)";
    $usuario=$_POST['usuario'];
    $apellido=$_POST['apellido'];
    $telefono=$_POST['telefono'];
    $correo=$_POST['correo'];
    $nombre=$_POST['nombre'];
    $contra=$_POST['contra'];

    if ($usuario!=""){
        if ($nombre!=""){
            if ($apellido!=""){
                if ($telefono!=""){
                    if ($correo!=""){
                        if ($contra!=""){

                            //encriptamos la contraseña
                            $contra=hash2( $contra);

                            $statement=$PDO->prepare($sql);
                            $statement->bindParam( parameter: ':usuario', &variable: $usuario);
                            $statement->bindParam( parameter: ':nombre', &variable: $nombre);
                            $statement->bindParam( parameter: ':apellido', &variable: $apellido);
                            $statement->bindParam( parameter: ':telefono', &variable: $telefono);
                            $statement->bindParam( parameter: ':correo', &variable: $correo);
                            $statement->bindParam( parameter: ':contra', &variable: $contra);

                            $statement->execute();

```

### 3.4.4 Sesiones y seguridad de usuarios.

Al verificar la autenticidad de una contraseña encriptada en la base de datos con el método password\_verify se guarda la id de la persona en una session que se podrá utilizar más adelante para hacer otras consultas.

```

if ($_SERVER['REQUEST_METHOD']=='POST') {
    $sql = "SELECT contra,id from propietario where usuario=:usuario";
    $usuario=$_POST['usuario'];
    $contra=$_POST['contra'];

    $statement=$PDO->prepare($sql);
    $statement->bindParam( parameter: ':usuario', &variable: $usuario);
    $statement->execute();
    $persona = $statement->fetch( fetch_style: PDO::FETCH_ASSOC);

    if (!$persona){
        //no hay usuario que se llame asi
        ?><script>
            alert("No se ha encontrado usuario con los datos indicados");
        </script>
        <?php
    }else if (password_verify($contra, $persona["contra"])){
        $_SESSION["id"]=$persona['id'];
        //las contraseñas coinciden y redirigimos a funciones de usuario
        header( string: 'Location:funcionesUsuario.php');
        exit();
    }else if(!password_verify($contra, $persona['contra'])){
        ?> <script>
            alert("La contraseña no coincide, vuelva a intentarlo");
        </script>
        <?php
    }
}

```

### 3.4.5 Conexión a base de datos en local

```
<?php
/** Created by PhpStorm. ...*/
class Connection
{
    public static function make(){
        try{
            $opciones=[PDO::MYSQL_ATTR_INIT_COMMAND=>"SET NAMES utf8",
                      PDO::ATTR_ERRMODE=>PDO::ERRMODE_EXCEPTION,
                      PDO::ATTR_PERSISTENT=>true];
            $conection=new PDO(
                dsn: 'mysql:localhost=127.0.0.1;dbname=inmobiliaria2;charset=utf8',
                username: 'root',
                passwd: '',
                $opciones);

            }catch (PDOException $PDOException){
                die($PDOException->getMessage());
            }
        return $conection;
    }
}
```

Nos conectamos a la base de datos en local con la clase Connection donde pondremos el nombre de la base de datos el usuario que tiene permiso para conectarse y la contraseña

Para poder utilizar la conexión tendremos que iniciar así cada página en la que queramos hacer consultas o insertar datos.

```
require_once 'Connection.php';
$PDO=Connection::make();
```

## 3.5 Diseño de interfaces y lenguaje de marcas.

### 3.5.1 Elección de estilos, colores, fuentes, planteamiento de diseño.

Se ha elegido como color principal y de marca el color azul. Este color es apropiado para empresas que **tratan de transmitir calma y confianza**.

Se ha decidido no sobrecargar la página con demasiados colores sino un estilo minimalista con imágenes y información que no sature la vista del cliente.

Se ha elegido dos tipos de fuente. Una para los títulos llamativa y una fuente clásica para la información para que sea bien legible.

```
<!-- fonts -->

<link href="https://fonts.googleapis.com/css?family=Pacifico" rel="stylesheet">
<link href="https://fonts.googleapis.com/css?family=Passion+One" rel="stylesheet">
<link href="https://fonts.googleapis.com/css?family=Lobster" rel="stylesheet">
<link href="https://fonts.googleapis.com/css?family=Anton" rel="stylesheet">
<!-- Bootstrap -->
<link href="css/bootstrap.min.css" rel="stylesheet">

<!-- Mis estilos -->
<link href="css/mi.css" rel="stylesheet">
```

### 3.5.2 Lenguaje de marcas y diseño de interfaces

#### 3.5.2.1 HTML

Para estructurar la aplicación web se ha usado HTML.

#### 3.5.2.2 CSS

Para darle estilos, colores y transiciones CSS.

```

    .zoom:hover {
        transform:scale(1.1, 1.1);
    }

    .zoomGrande:hover {
        transform:scale(1.3, 1.3);
    }

    .zoomMuyGrande:hover {
        transform:scale(1.5, 1.5);
    }

    .helicoptero:hover {
        transform: rotate(360deg);
        transition: 3s;
    }

    h1 {
        font-family: 'Pacifico', cursive;
    }

    .circulito:hover{
        border-radius: 50%;
    }

    .contraste:hover{
        filter:contrast(200%);
    }

    .arcoiris:hover{
        animation-name: colores;
        animation-duration: 10s;
    }

    @keyframes colores {
        0%   {background-color: yellow;}
        25%  {background-color: greenyellow;}
        50%  {background-color: deepskyblue;}
        100% {background-color: red;}
    }

    .derecha{
        position:absolute;
        right:10px;
    }

```

### 3.5.2.3 Bootstrap

Se ha usado el framework de bootstrap para que la página web sea responsive. Por parte de los profesores se me sugirió la utilización de tarjetas de bootstrap lo cual fue muy útil a la hora de exponer la información de los inmuebles. Bootstrap se divide hasta 12 columnas. En este caso tenemos una tarjeta que tendrá un tamaño de lg 3 que serán 4 tarjetas, en md-6 serían 2 tarjetas de 6 y más pequeño que eso sería el col por defecto.

```
<h1 class="text-center ">A continuacion nuestra mejor seleccion de inmuebles</h1>
<div class="row text-center">
    <?php foreach($objetos as $objeto): ?>
        <div class="col-lg-3 col-md-6">

            <div class="card h-100">
                <img class="card-img-top zoom" src=<?php echo $objeto['foto1'];?>>
                <div class="card-body">
                    <h4 class="card-title">Ciudad <?php echo $objeto['ciudad'];?> </h4>
                    <p class="card-text">
                        Informacion sobre inmueble:
                    </p>
                    <p class="text-left card-text">
                        Tipo de inmueble: <?php echo $objeto['tipo'];?><br>
                        Habitaciones: <?php echo $objeto['habitaciones'];?><br>
                        Metros cuadrados: <?php echo $objeto['metros'];?><br>
                        Altura: <?php echo $objeto['piso'];?><br>
                        Numero de referencia:
                    </p>
                    <form action="masInformacion.php" method="POST" enctype="multipart/form-data">
                        <input type="text" readonly name="id" value=<?php echo $objeto['id'];?> />
                    </form>
                </div>
                <div class="card-footer">
                    <!--class="btn btn-primary submit"-->
                    <input class="btn btn-primary" type="submit" value="Mas informacion"></input>
                </div>
            </div>
        </div>
    <?php endforeach; ?>
```

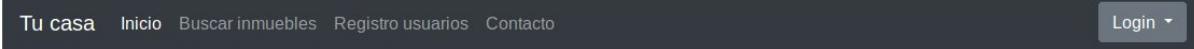
También se ve en la imagen la utilización de estilos de colores de botones, por ejemplo primary en este caso es botón azul. El alineamiento del texto, la posición de la imagen con las clases por defecto mas las mias propias añadidas después.

### 3.5.2.4 Gimp

Se ha utilizado lo aprendido durante el curso para crear y editar imágenes con la herramienta gimp. Se han creado imágenes que facilitan la accesibilidad de volver atrás fácilmente y no tener que recorrer la barra de navegación y evitar así que el usuario tenga que dar varios click para volver al sitio deseado y hacer por ejemplo una nueva búsqueda.



Se ha diseñado una barra de navegación muy sencilla y se ha evitado que se sature el usuario con información.



## 3.6 Entornos de desarrollo.

### 3.6.1 Metodología de desarrollo del software

Se ha utilizado el método de desarrollo en cascada. Analizando las necesidades de la aplicación pasando al diseño e implementación. Cuando se detectaba una carencia se volvió atrás para poder mejorar y solucionar el problema. En mi caso he tenido que rehacer la barra de navegación varias veces, he tenido que añadir más páginas con diferentes funciones que no tenía pensadas de un principio.



Despliegue aplicación.

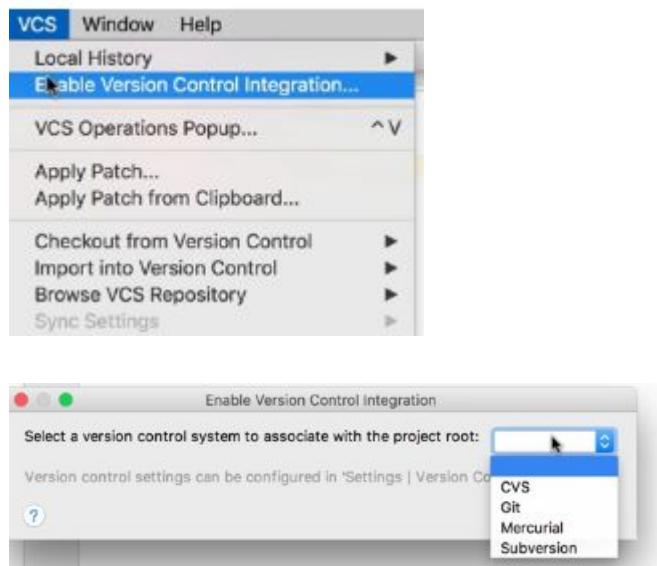
### 3.6.2 Sistema de gestión de versiones

Se ha utilizado el sistema de gestión de versiones gratuito github.

<https://github.com/>

Hemos configurado php storm para que se pueda subir directamente desde ahí.

Desde php storm le damos a habilitar control de versiones y seleccionamos git



Al hacer esto se nos creará un nuevo repositorio git y podremos hacer los siguientes comandos:

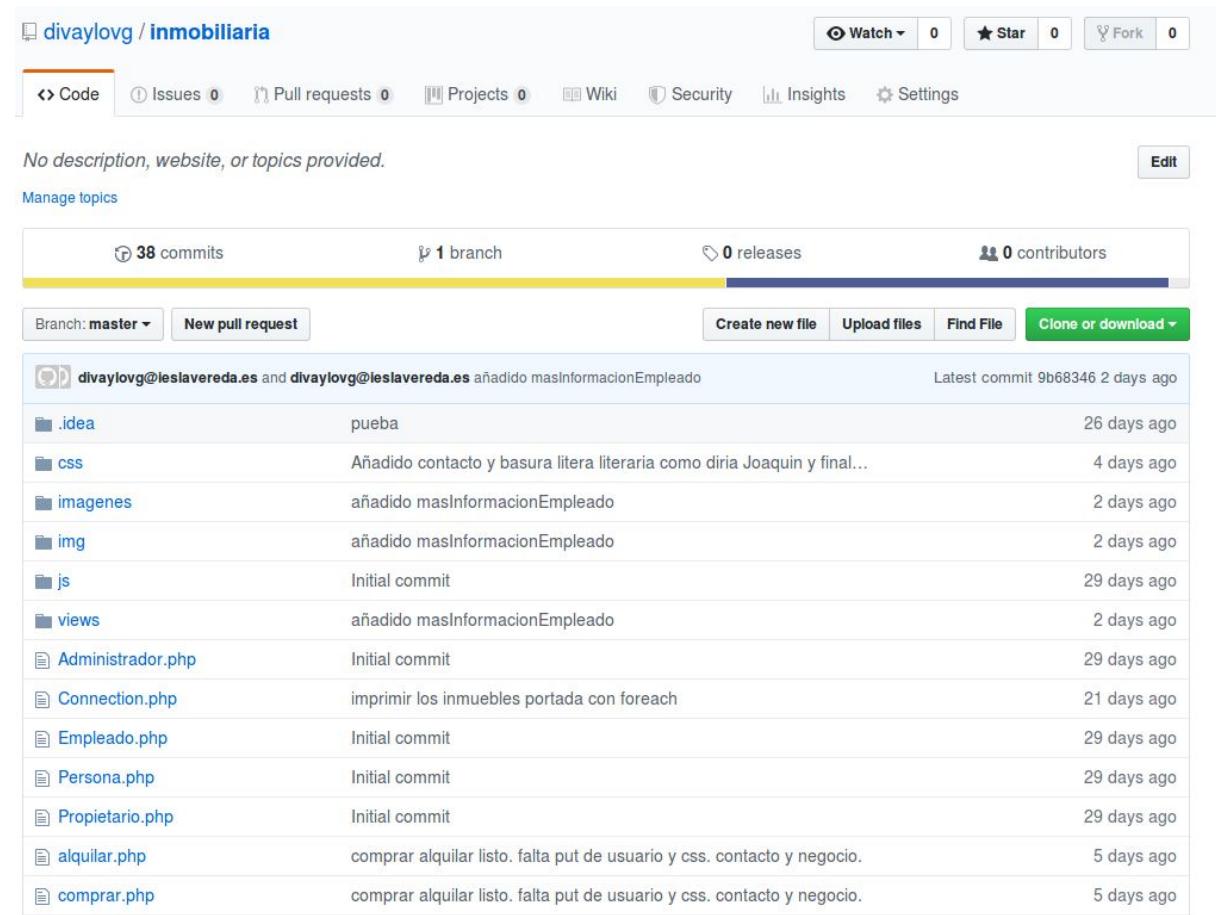
Con control K hacemos commit y con control mayúscula K hacemos push,

Aquí está mi repositorio en el cual se puede descargar mi proyecto.

<https://github.com/divaylovg/inmobiliaria>

Y aquí podemos ver los cambios que se han hecho y clonar o bajar nuestro repositorio.

### 3.6.2 Sistema de gestión de versiones



The screenshot shows a GitHub repository page for 'divaylovg / inmobiliaria'. The repository has 38 commits, 1 branch, 0 releases, and 0 contributors. The latest commit was 9b68346, 2 days ago. The commit history lists various file changes and their descriptions:

File	Description	Date
.idea	pueba	26 days ago
css	Añadido contacto y basura literaria como diria Joaquin y final...	4 days ago
imagenes	añadido masInformacionEmpleado	2 days ago
img	añadido masInformacionEmpleado	2 days ago
js	Initial commit	29 days ago
views	añadido masInformacionEmpleado	2 days ago
Administrador.php	Initial commit	29 days ago
Connection.php	imprimir los inmuebles portada con foreach	21 days ago
Empleado.php	Initial commit	29 days ago
Persona.php	Initial commit	29 days ago
Propietario.php	Initial commit	29 days ago
alquilar.php	comprar alquilar listo. falta put de usuario y css. contacto y negocio.	5 days ago
comprar.php	comprar alquilar listo. falta put de usuario y css. contacto y negocio.	5 days ago

### 3.6.3 Plan de pruebas. Caja blanca

Se han hecho pruebas introduciendo datos con el método de la caja blanca que consiste en meter datos sin saber cuál será el resultado para encontrar errores. Ya que se sabía desde un principio que aquí es el momento clave de la aplicación y es donde más se nos pasan cosas por alto, se ha intentado evitar que se puedan meter demasiados datos en algunas páginas que no sean formularios. Por ejemplo aquí en la búsqueda de alquiler o venta el único campo en el que se puede meter información es en el campo del dinero. Sin embargo estamos introduciendo 4 variables diferentes y tenemos controladas 3 de ellas que serán si o si correctas. En el caso del dinero lo hemos controlado con php

Tu casa Inicio Buscar inmuebles Registro usuarios Contacto Login ▾



Dinero

Cuanto es tu presupuesto maximo?



Tipo de vivienda

Elige tipo de vivienda que te interesa.

chalet ▾



Valencia

Ciudad

Elige la ciudad en la que deseas vivir.

Valencia ▾



Habitaciones

Numero minimo de habitaciones:

Sin habitaciones ▾

[Buscar](#)

Si nos devuelve null en vez de imprimir y que nos salga error vamos a ver si tenemos objetos. Si no los hay nos redirigirá a location: comprar o alquilar de nuevo para volver a introducir los datos.

```
if ($_SERVER['REQUEST_METHOD']=='POST') {
    //SELECT * FROM inmueble where alquiler< 5000 and tipo = 'chalet' and ciudad = 'Valencia' and habitaciones< 4
    $dinero=$_POST['dinero'];
    $tipo=$_POST['tipo'];
    $ciudad=$_POST['ciudad'];
    $habitaciones=$_POST['habitaciones'];

    $statement=$PDO->prepare( statement: "SELECT * FROM inmueble where venta<:dinero and venta>0 and tipo = :tipo and ciudad = :ciudad and habitaciones>= :habitaciones");
    $statement->setFetchMode( mode: PDO::FETCH_CLASS|PDO::FETCH_PROPS_LATE, classNameObject: "inmueble");
    $statement->bindParam( parameter: ':dinero', &variable: $dinero);
    $statement->bindParam( parameter: ':tipo', &variable: $tipo);
    $statement->bindParam( parameter: ':ciudad', &variable: $ciudad);
    $statement->bindParam( parameter: ':habitaciones', &variable: $habitaciones);
    $statement->execute();
    $objetos=$statement->fetchAll();
    // print_r($objetos);

    if (!$objetos) {
        //no hay objetos redirigimos atras
        header( string: 'Location:comprar.php');
    }
}
```

En otros lugares con más datos se nos ha pasado por alto por ejemplo el caso de clave primaria duplicada. Se tomará nota y en un futuro se podrá corregir con javascript controlando el envío de datos o con php haciendo un select de todos los usuarios que hay en la base de datos, recorriendo y comparando los nombres.

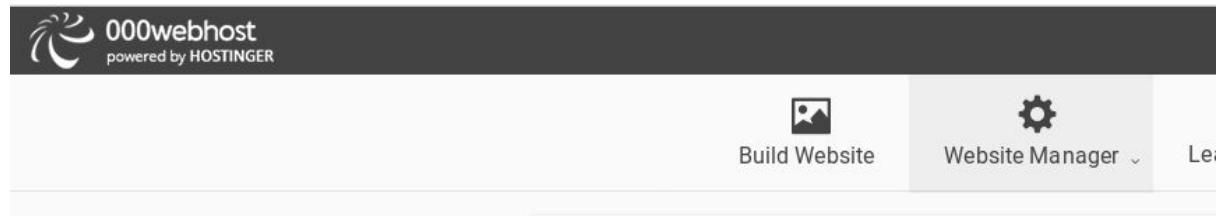
**Fatal error:** Uncaught PDOException: SQLSTATE[23000]: Integrity constraint violation:  
 1062 Duplicate entry 'Tzar' for key 'usuario'  
 /opt/lampp/htdocs/inmobiliaria/registrarUsuario.php:36 Stack trace: #0  
 /opt/lampp/htdocs/inmobiliaria/registrarUsuario.php(36): PDOStatement->execute() #1  
 {main} thrown in **/opt/lampp/htdocs/inmobiliaria/registrarUsuario.php** on line **36**

### 3.7 Despliegue de aplicaciones

Se ha visto el servidor local y como instalarlo. Ahora vamos a ver el despliegue en un servidor remoto.

<https://www.000webhost.com>

Te permite subir tus páginas web de forma gratuita con ciertas limitaciones. Antes vimos como subir la base de datos al mismo dominio ahora veremos el despliegue del resto de la aplicación.



### Vamos a la pestaña de file manager y upload files



### Fast, Easy & Secure File Upload

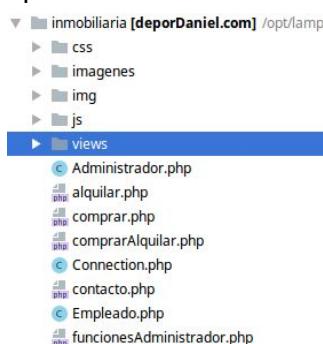
Click the button below to access file manager.

[Upload files now](#)

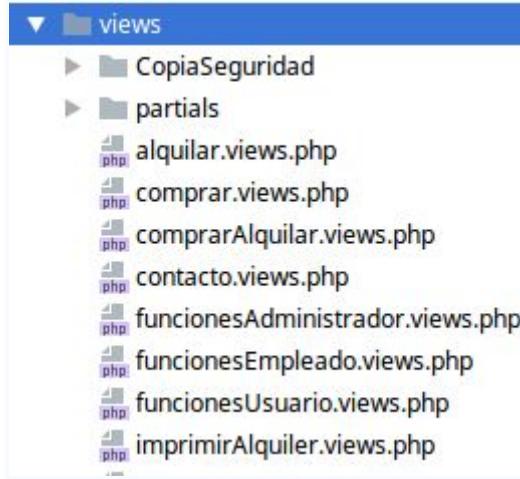
Nuestra página web está estructurada en varias capas. En la primera están las carpetas de imágenes-las imágenes que suben los usuarios  
css- están los estilos de bootstrap y nuestros propios  
img-los que vamos a utilizar para nuestra página  
js- los javascript de bootstrap

Las páginas principales de php están alojadas también aquí. Son los que contienen la parte lógica de programación. También se alojan las clases de objeto.

En la carpeta de views están los php



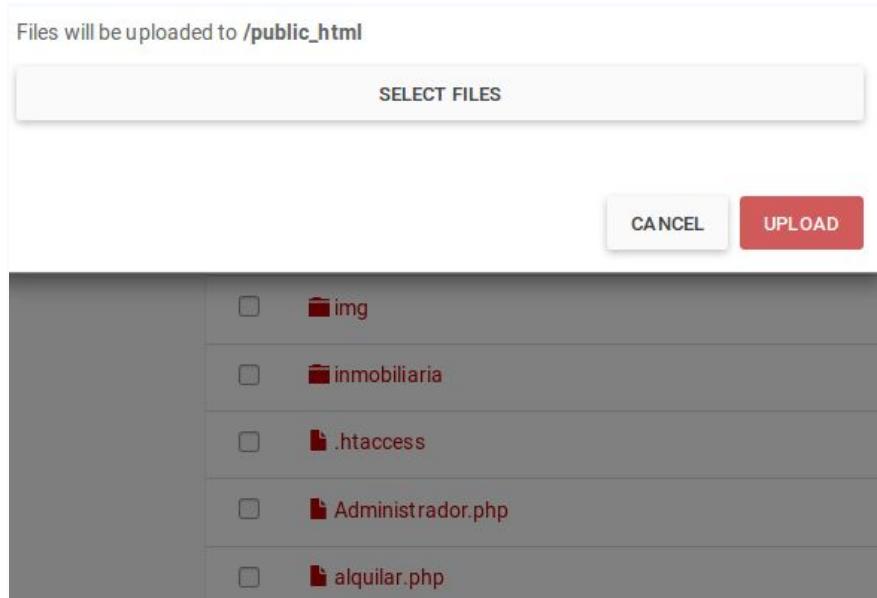
A su vez los php llamarán a los documentos de views. Como el nombre lo indica será la parte visual del php. Allí estarán los formularios que llamarán a las páginas de php donde estará la lógica para procesar los datos. Para llamar a estos ficheros utilizamos `require "views/index.views.php";`



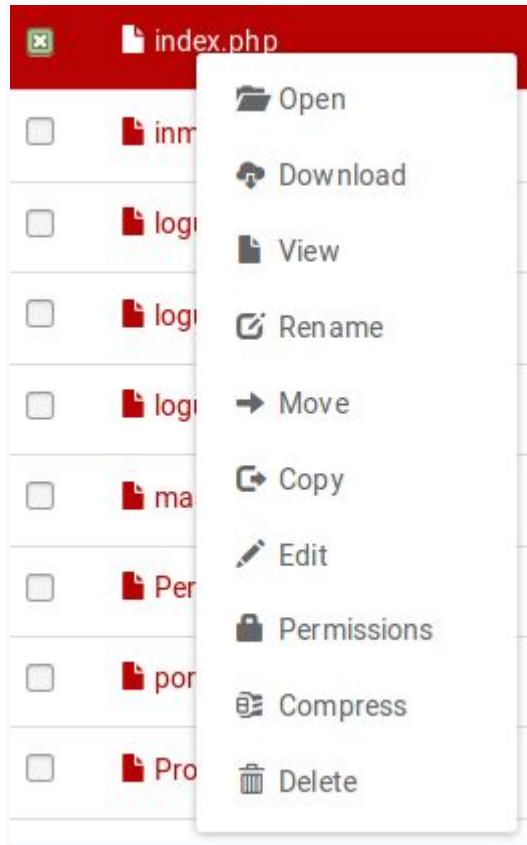
A su vez estos llaman al documento de la carpeta partials en el que están todos los elementos que serán comunes a toda la página. Por ejemplo la barra de navegación que está fija y en vez de escribir 30 veces el mismo código podemos poner un archivo que será común a todos.

`require __DIR__."/partials/inmobiliaria.part.php";`

Ahora sabiendo la estructura del proyecto solo tenemos que crear las mismas carpetas en el servidor remoto y subir los archivos que tenemos en el proyecto de la misma forma.



Podemos editar descargar ver etc todos los archivos



Esta es la url de nuestra página.

<https://divaylovg.000webhostapp.com/inmobiliaria/index.php>

### 3.8 Entorno cliente.

```
<div class="form-row">

    <div class="col-lg-3 col-md-6 mb-4">
        <label for="validationTooltip01">Precio alquiler</label>
        <input type="text" name="alquiler" onblur="comprobarNumerosAlquiler();" placeholder="Entero" >
    </div>

    <div class="col-lg-3 col-md-6 mb-4">
        <label for="validationTooltip01">Precio venta</label>
        <input type="text" name="venta" onblur="comprobarNumerosVenta();" placeholder="Entero" >
    </div>

    <div class="col-lg-3 col-md-6 mb-4">
        <label for="validationTooltip01">Metros cuadrados</label>
        <input type="text" class="form-control" onblur="comprobarNumerosMetros();" name="metros" placeholder="Metros" required>
    </div>

    <div class="col-lg-3 col-md-6 mb-4">
        <label for="validationTooltip01">Calle</label>
        <input type="text" class="form-control" name="calle" onblur="comprobarLetrasCalle();" placeholder="Nombre de la calle" required>
    </div>

    <div class="col-lg-3 col-md-6 mb-4">
        <label for="validationTooltip01">Número</label>
        <input type="text" class="form-control" name="numero" onblur="comprobarNumeroCalle();" placeholder="Entero" required>
    </div>
```

De cara al cliente se ha utilizado el lenguaje de programación JavaScript

Se han usado eventos como por ejemplo onblur para validar los formularios. En este caso si los datos introducidos serán comprobadas con expresiones regulares para indicar al usuario que son correctas o no.

```
<SCRIPT LANGUAGE="Javascript">
let arCampos=document.getElementsByTagName('input');

let regNumero= /^\\d+$/;
let regLetras= /^\\D+$/;

function comprobarNumerosAlquiler(){
    if (regNumero.test(arCampos[0].value)) {
        arCampos[0].style.background="green";
    }else{
        arCampos[0].style.background="red";
    }
}

function comprobarNumerosVenta(){
    if (regNumero.test(arCampos[1].value)) {
        arCampos[1].style.background="green";
    }else{
        arCampos[1].style.background="red";
    }
}

function comprobarNumerosMetros(){
    if (regNumero.test(arCampos[2].value)) {
        arCampos[2].style.background="green";
    }else{
        arCampos[2].style.background="red";
    }
}
```

## 4 Problemas

---

Inserción de varias imágenes.

Problema editar auto incremental en base datos por foreign key con herramienta visual, se ha hecho con sql.

Hacer que sean objetos y crear con javascript la tarjeta de cada objeto.

Como y donde hacer el dom para meter los objetos? al final se ha solucionado haciéndolo con php que ha resultado mucho más fácil.

En alquiler y venta no se me insertan si no meto nada y tiene por defecto null en la base de datos, me da problemas. No se si es de la base de datos o de la programación php  
//SOLUCIONADO con isset

pendiente solucionar problemas de plan de pruebas. Caja blanca

Al insertar fotos se insertan bien pero si no se insertan todas las imagenes falla.

## 5 Mejorar la aplicación

---

Que los empleados también puedan dar de alta inmuebles.

Seguridad mediante javascript o sql para entrar en crear administradores.

Filtrar mejor los datos y poner excepciones si se repite en la base de datos.

Poner tildes en texto.

## 6 El código, mejor descargar desde git

---

<https://github.com/divaylovg/inmobiliaria>

```
inmobiliaria.part.php
<!DOCTYPE html>
<html lang="en">

<head>

<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
<title>Inmobiliaria Tu Casa</title>

<!-- fonts -->

<link href="https://fonts.googleapis.com/css?family=Pacifico" rel="stylesheet">
<link href="https://fonts.googleapis.com/css?family=Passion+One" rel="stylesheet">
<link href="https://fonts.googleapis.com/css?family=Lobster" rel="stylesheet">
<link href="https://fonts.googleapis.com/css?family=Anton" rel="stylesheet">
<!-- Bootstrap -->
<link href="css/bootstrap.min.css" rel="stylesheet">

<!-- Mis estilos -->
<link href="css/mi.css" rel="stylesheet">

</head>

<body>
```

```

<div class="container">
    <div class="row text-center">
        
    </div>

    <!-- Barra navegacion -->
    <nav class="navbar navbar-expand-lg navbar-dark bg-dark">
        <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarsExample05"
            aria-controls="navbarsExample05" aria-expanded="false" aria-label="Toggle navigation">
            <span class="navbar-toggler-icon"></span>
        </button>

        <div class="collapse navbar-collapse" id="navbarsExample05">
            <ul class="navbar-nav ">
                <a class="navbar-brand align-self-lg-end" href="index.php">Tu casa</a>

                <li class="nav-item active">
                    <a class="nav-link" href="index.php">Inicio <span class="sr-only">(current)</span></a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="comprarAlquilar.php">Buscar inmuebles</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="registrarUsuario.php">Registro usuarios</a>
                </li>

                <li class="nav-item">
                    <a class="nav-link" href="contacto.php">Contacto</a>
                </li>

                <div class="dropdown float-right derecha">
                    <button class="btn btn-secondary dropdown-toggle" type="button" id="dropdownMenuButton"
                        data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
                        Login
                    </button>
                    <div class="dropdown-menu" aria-labelledby="dropdownMenuButton">
                        <a class="dropdown-item" href="loguearUsuario.php">Entrar Usuario</a>
                        <a class="dropdown-item" href="loguearEmpleado.php">Entrar Empleado</a>
                        <a class="dropdown-item" href="loguearAdministrador.php">Entrar Administrador</a>
                    </div>
                </div>
            </ul>
        </div>
    </nav>

```

```

alquilar.views.php
<?php
ob_start();

require __DIR__."/partials/inmobiliaria.part.php";
require_once 'Connection.php';
$PDO=Connection::make();

```

```

if ($_SERVER['REQUEST_METHOD']=='GET') {
    //obtenemos todos los inmuebles agrupando por ciudad para que no nos salga mas de una vez las repetidas
    $stmt=$PDO->prepare("SELECT * FROM inmueble group by ciudad");
    $stmt->setFetchMode(PDO::FETCH_CLASS|PDO::FETCH_PROPS_LATE,"inmueble");
    $stmt->execute();
    $ciudades=$stmt->fetchAll();

    //obtenemos todos los inmuebles por tipo y agrupando para que no se repita
    $stmt=$PDO->prepare("SELECT * FROM inmueble group by tipo");
    $stmt->setFetchMode(PDO::FETCH_CLASS|PDO::FETCH_PROPS_LATE,"inmueble");
    $stmt->execute();
    $tipos=$stmt->fetchAll();
}

?>
<div class="container">
    <form action="imprimirAlquiler.php" method="POST" enctype="multipart/form-data">

        <div class="row text-center">
            <div class="col-lg-3 col-md-6 mb-4 zoom">
                <div class="card h-100">
                    
                    <div class="card-body">
                        <h4 class="card-title">Dinero </h4>
                        <p class="card-text">
                            Cuanto es tu presupuesto máximo?
                        </p>
                        <input type="text" name="dinero" id="dinero" value="" required>
                    </div>
                </div>
            </div>
        </div>

        <div class="col-lg-3 col-md-6 mb-4 zoom">
            <div class="card h-100">
                
                <div class="card-body">
                    <h4 class="card-title">Tipo de vivienda </h4>
                    <p class="card-text">
                        Elige tipo de vivienda que te interesa.
                    </p>
                    <select name="tipo">
                        <?php foreach($tipos as $tipo): ?>
                            <option value=<?php echo $tipo['tipo']?> name="tipo"><?php echo $tipo['tipo'] ?></option>;
                        <?php endforeach; ?>
                    </select>
                </div>
            </div>
        </div>

        <div class="col-lg-3 col-md-6 mb-4 zoom">
            <div class="card h-100">
                
                <div class="card-body">
                    <h4 class="card-title">Ciudad </h4>
                    <p class="card-text">
                        Elige la ciudad en la que deseas vivir.
                    </p>
                </div>
            </div>
        </div>
    </form>

```

```

        </p>
        <select name="ciudad">
            <?php foreach($ciudades as $ciudad): ?>
                <option value=<?php echo $ciudad['ciudad']?> name="ciudad"><?php echo $ciudad['ciudad']?
            ?></option>;
        <?php endforeach; ?>
        </select>
    </div>
</div>
</div>

<div class="col-lg-3 col-md-6 mb-4 zoom">
    <div class="card h-100">
        
        <div class="card-body">
            <h4 class="card-title">Habitaciones </h4>
            <p class="card-text">
                Número mínimo de habitaciones:
            </p>
            <select class="custom-select my-1 mr-sm-2" name="habitaciones" id="habitaciones">
                <option value="0">Sin habitaciones<option>
                <option value="1">1</option>
                <option value="2">2</option>
                <option value="3">3</option>
                <option value="4">4</option>
                <option value="5">5</option>
                <option value="6">6</option>
                <option value="7">7</option>
                <option value="8">8</option>
                <option value="9">9</option>
                <option value="10">10</option>
            </select>
        </div>
    </div>
</div>
<div class="card-footer col">
    <input class="btn btn-primary" type="submit" value="Buscar"></input>
</div>
</form>
</div>

<!-- Footer -->
<footer class="py-1 bg-dark ">
    <div class="container">
        <div class="row">
            <div class="col-3">
                <p class="text-primary "> Daniel Ivaylov</p>
            </div>
            <div class="col-3">
                <a href="index.php"><button type="button" class="btn btn-primary">Inicio</button></a>
            </div>
            <div class="col-3">
                <p class="text-primary ">Inmobiliaria en: C/San Sebastian 5 Valencia CP 46019</p>
            </div>
        </div>
    </div>

```

```

<div class="col-3">
    <marquee class="text-primary" behavior="scroll" direction="left"> Información al 961 616 161 </marquee>
</div>

</div>
</div>
</body>

</html>

comprar.views.php
<?php
ob_start();

require __DIR__."/partials/inmobiliaria.part.php";
require_once 'Connection.php';
$PDO=Connection::make();

if ($_SERVER['REQUEST_METHOD']=='GET') {
    //obtenemos todos los inmuebles agrupando por ciudad para que no nos salga mas de una vez las repetidas
    $stmt=$PDO->prepare("SELECT * FROM inmueble group by ciudad");
    $stmt->setFetchMode(PDO::FETCH_CLASS|PDO::FETCH_PROPS_LATE,"inmueble");
    $stmt->execute();
    $ciudades=$stmt->fetchAll();

    //obtenemos todos los inmuebles por tipo y agrupando para que no se repita
    $stmt=$PDO->prepare("SELECT * FROM inmueble group by tipo");
    $stmt->setFetchMode(PDO::FETCH_CLASS|PDO::FETCH_PROPS_LATE,"inmueble");
    $stmt->execute();
    $tipos=$stmt->fetchAll();
}

?>
<div class="container">
    <form action="imprimirComprar.php" method="POST" enctype="multipart/form-data">

        <div class="row text-center">
            <div class="col-lg-3 col-md-6 mb-4 zoom">
                <div class="card h-100">
                    
                    <div class="card-body">
                        <h4 class="card-title">Dinero </h4>
                        <p class="card-text">
                            Presupuesto máximo para comprar
                        </p>
                        <input type="text" name="dinero" id="dinero" value="" required>
                
```

```

        </div>
    </div>
</div>

<div class="col-lg-3 col-md-6 mb-4 zoom">
    <div class="card h-100">
        
        <div class="card-body">
            <h4 class="card-title">Tipo de vivienda </h4>
            <p class="card-text">
                Elegir tipo vivienda para comprar
            </p>
            <select name="tipo">
                <?php foreach($tipos as $tipo): ?>
                    <option value=<?php echo $tipo['tipo']?> name="tipo"><?php echo $tipo['tipo'] ?></option>;
                <?php endforeach; ?>
            </select>
        </div>
    </div>
</div>

```

```

<div class="col-lg-3 col-md-6 mb-4 zoom">
    <div class="card h-100">
        
        <div class="card-body">
            <h4 class="card-title">Ciudad </h4>
            <p class="card-text">
                Elige la ciudad en la que deseas vivir.
            </p>
            <select name="ciudad">
                <?php foreach($ciudades as $ciudad): ?>
                    <option value=<?php echo $ciudad['ciudad']?> name="ciudad"><?php echo $ciudad['ciudad'] ?>
                ?></option>;
                <?php endforeach; ?>
            </select>
        </div>
    </div>
</div>

```

```

<div class="col-lg-3 col-md-6 mb-4 zoom">
    <div class="card h-100">
        
        <div class="card-body">
            <h4 class="card-title">Habitaciones </h4>
            <p class="card-text">
                Número mínimo de habitaciones:
            </p>
            <select class="custom-select my-1 mr-sm-2" name="habitaciones" id="habitaciones">
                <option value="0">Sin habitaciones<option>
                <option value="1">1</option>
                <option value="2">2</option>
                <option value="3">3</option>
                <option value="4">4</option>
                <option value="5">5</option>
                <option value="6">6</option>
                <option value="7">7</option>
                <option value="8">8</option>
            </select>
        </div>
    </div>
</div>

```

```

        <option value="9">9</option>
        <option value="10">10</option>
    </select>
</div>

</div>
</div>
<div class="card-footer col">
    <input class="btn btn-primary" type="submit" value="Buscar"></input>
</div>
</form>
</div>

<!-- Footer -->
<footer class="py-1 bg-dark ">
    <div class="container">
        <div class="row">
            <div class="col-3">
                <p class="text-primary "> Daniel Ivaylov</p>
            </div>

            <div class="col-3">
                <a href="index.php"><button type="button" class="btn btn-primary">Inicio</button></a>
            </div>

            <div class="col-3">
                <p class="text-primary ">Inmobiliaria en: C/San Sebastian 5 Valencia CP 46019</p>
            </div>

            <div class="col-3">
                <marquee class="text-primary " behavior="scroll" direction="left"> Información al 961 616 161 </marquee>
            </div>
        </div>
    </div>
</footer>

</div>
<!-- Bootstrap JavaScript -->
<script src="js/jquery-3.3.1.min.js"></script>
<script src="js/bootstrap.js"></script>

</div>
</body>

</html>

comprarAlquilar.views.php
```

```

<?php
ob_start();

require __DIR__."/partials/inmobiliaria.part.php";
require_once 'Connection.php';
$PDO=Connection::make();
```

```

if ($_SERVER['REQUEST_METHOD']=='GET') {

$stmt=$PDO->prepare("SELECT * FROM inmueble group by ciudad");
$stmt->setFetchMode(PDO::FETCH_CLASS|PDO::FETCH_PROPS_LATE,"inmueble");
$stmt->execute();
$ciudades=$stmt->fetchAll();
}

?>

<div class="container m-5">

<div class="row justify-content-around ">
<div class="col-4 ">
<a title="Vender" href="alquilar.php">

</div>
<div class="col-4 ">
<a title="Vender" href="comprar.php">

</div>
</div>
</div>

<!-- Footer -->
<footer class="py-1 bg-dark ">
<div class="container">
<div class="row">
<div class="col-3">
<p class="text-primary "> Daniel Ivaylov</p>
</div>

<div class="col-3">
<a href="index.php"><button type="button" class="btn btn-primary">Inicio</button></a>
</div>

<div class="col-3">
<p class="text-primary ">Inmobiliaria en: C/San Sebastian 5 Valencia CP 46019</p>
</div>

<div class="col-3">
<marquee class="text-primary " behavior="scroll" direction="left"> Información al 961 616 161 </marquee>
</div>

</div>
</div>
</footer>

</div>
<!-- Bootstrap JavaScript -->
<script src="js/jquery-3.3.1.min.js"></script>
<script src="js/bootstrap.js"></script>

</div>
</body>

```

```

</html>
contacto.views.php
<?php
ob_start();

require __DIR__."/partials/inmobiliaria.part.php";
require_once 'Connection.php';
$PDO=Connection::make();

$objetos=[];

if ($_SERVER["REQUEST_METHOD"]=="GET") {

$stmt=$PDO->prepare("SELECT * FROM inmueble");
$stmt->setFetchMode(PDO::FETCH_CLASS|PDO::FETCH_PROPS_LATE,"inmueble");
$stmt->execute();
$objetos=$stmt->fetchAll();
}

?>

<div class="container">

<div class="row align-items-center my-5 contraste zoom">
<div class="col-lg-7">

</div>
<div class="col-lg-5">
<h1 class="font-weight-light">Inmobiliaria tu casa</h1>
<p>El <b>valor añadido</b> de nuestra empresa consiste en que no existe la palabra permanencia. Usted no se atara a través de un contrato de exclusividad con nosotros como hacen todas las demás inmobiliarias. Confiamos tanto en la <b>profesionalidad de nuestros agentes </b>que estamos seguros de que su inmueble se venderá/alquilará rápido y si no es así usted mismo puede dar de baja el inmueble que ha registrado en nuestra página web. </p>
</div>
</div>

<table class="table table-striped">
<h1>Donde nos encontramos.</h1>
<tbody>
<tr class="fila">
<th scope="row">Ciudad:</th>
<td>Valencia</td>
<th>CP:</th>
<td>46019 </td>
</tr>

<tr class="fila">
<th scope="row">Calle:</th>
<td>C/San Sebastian 5 </td>
<th>Telefono:</th>
<td>961 616 161</td>
</tr>
<tr class="fila">
<th scope="row">Email:</th>
<td>tucasa@ieslavereda.es </td>
<th>Responsable</th>

```

```

<td>Daniel Ivaylov Gerchev</td>
</tr>
</tbody>
</table>

<div class="card text-white bg-primary mb-5 text-center">
    <div class="card-body">
        <p class="text-white ">Le ofrecemos asesoramiento, visitas guiadas por nuestros agentes a clientes y tasación totalmente gratuitos. No le cobraremos ni un céntimo hasta consigamos vender/alquilar su inmueble. No tendrá que lidiar con usuarios molestos ni personas curiosas. </p>
    </div>
</div>

<div class="row">
    <div class="col-md-6 mb-5">
        <div class="card h-100 arcoiris">
            <div class="card-body">
                <h2 class="card-title">Registrar usuario</h2>
                <p class="card-text">Si se registra como usuario podrá gestionar todos sus inmuebles, para alquilar o vender online, de manera muy sencilla. Quiere dar de alta un inmueble En tan sólo 30 segundos ya estará subido online y la gente puede recibir toda la información sobre el.</p>
            </div>
            <div class="card-footer">
                <a href="registrarUsuario.php" class="btn btn-primary btn-sm">Registrarse</a>
            </div>
        </div>
    </div>
    <div class="col-md-6 mb-5 ">
        <div class="card h-100 arcoiris">
            <div class="card-body">
                <h2 class="card-title">Buscar inmuebles</h2>
                <p class="card-text">Puede buscar inmuebles para comprar y alquilar por toda España. Tan fácil como filtrar por el dinero máximo que desea gastarse, la ciudad en la que desea vivir, el número de habitaciones y el tipo de inmueble.</p>
            </div>
            <div class="card-footer">
                <a href="comprarAlquilar.php" class="btn btn-primary btn-sm">Buscar</a>
            </div>
        </div>
    </div>
</div>

</div>

<!-- Footer -->
<footer class="py-1 bg-dark ">
    <div class="container">
        <div class="row">
            <div class="col-3">
                <p class="text-primary "> Daniel Ivaylov</p>
            </div>
            <div class="col-3">
                <a href="index.php"><button type="button" class="btn btn-primary">Inicio</button></a>
            </div>
            <div class="col-3">

```

```

        <p class="text-primary ">Inmobiliaria en: C/San Sebastian 5 Valencia CP 46019</p>
    </div>

    <div class="col-3">
        <marquee class="text-primary " behavior="scroll" direction="left"> Información al 961 616 161 </marquee>
    </div>

    </div>
</div>
</footer>

<!-- Bootstrap JavaScript -->
<script src="js/jquery-3.3.1.min.js"></script>
<script src="js/bootstrap.js"></script>

</div>
</body>

</html>

```

```

funcionesAdministrador.views.php
<?php
require __DIR__ . "/partials/inmobiliaria.part.php";

ob_start();

?>

<div class="container col 12 m-5">
    <div class="row text-left">

        <form action="funcionesAdministrador.php" method="POST" enctype="multipart/form-data">
            <h1>Registra a tus empleados para que puedan vender, alquilar y borrar pisos.</h1>

            <div class="form-group">
                <label class="text-left" for="formGroupExampleInput">Usuario</label>
                <input type="text" class="form-control" id="nombre" name="usuario" placeholder="Ejemplo: TzarDaniel"
required>
            </div>

            <div class="form-group">
                <label class="text-left" for="formGroupExampleInput">Nombre</label>
                <input type="text" class="form-control" id="nombre" onblur="comprobarLetrasNombre();" name="nombre"
placeholder="Ejemplo: Daniel" required>
            </div>

            <div class="form-group">
                <label class="text-left" for="formGroupExampleInput2">Apellido</label>
                <input type="text" class="form-control" id="apellido" onblur="comprobarLetrasApellido();" name="apellido"
placeholder="Ejemplo: Gerchev" required>
            </div>

            <div class="form-group">
                <label class="text-left" for="formGroupExampleInput2">Telefono</label>
                <input type="text" class="form-control" id="telefono" onblur="comprobarTelefono();" name="telefono"
placeholder="Ejemplo:653366987" required>
            </div>

```

```

<div class="form-group">
    <label class="text-left" for="exampleInputEmail1">Email</label>
    <input type="email" class="form-control" aria-describedby="emailHelp" name="correo"
placeholder="Ejemplo: divaylov@ieslavereda.es" required>
    <small class="form-text text-muted">Nunca vamos a dar su informacion a terceros.</small>
</div>
<div class="form-group">
    <label class="text-left" for="exampleInputPassword1">Contraseña</label>
    <input type="password" class="form-control" name="contra" placeholder="Introduce contraseña" required>
</div>

    <input type="submit" value="Enviar">
</form>

</div>
</div>

<SCRIPT LANGUAGE="Javascript">
let arCampos=document.getElementsByTagName('input');

let regNumero=/^\d+$/;
let regLetras=/^D+$/;

function comprobarLetrasNombre(){
    if (regLetras.test(arCampos[1].value)) {
        arCampos[1].style.background="green";
    }else{
        arCampos[1].style.background="red";
    }
}

function comprobarLetrasApellido(){
    if (regLetras.test(arCampos[2].value)) {
        arCampos[2].style.background="green";
    }else{
        arCampos[2].style.background="red";
    }
}

function comprobarTelefono(){
    if (regNumero.test(arCampos[3].value)) {
        arCampos[3].style.background="green";
    }else{
        arCampos[3].style.background="red";
    }
}

</SCRIPT>

<!-- Footer -->
<footer class="py-1 bg-dark ">
    <div class="container">
        <div class="row">
            <div class="col-3">
                <p class="text-primary "> Daniel Ivaylov</p>
            </div>
            <div class="col-3">

```

```

        <a href="index.php"><button type="button" class="btn btn-primary">Inicio</button></a>
    </div>

    <div class="col-3">
        <p class="text-primary ">Inmobiliaria en: C/San Sebastian 5 Valencia CP 46019</p>
    </div>

    <div class="col-3">
        <marquee class="text-primary " behavior="scroll" direction="left"> Informacion al 961 616 161 </marquee>
    </div>

    </div>
</div>
</footer>

<!-- Bootstrap JavaScript -->
<script src="js/jquery-3.3.1.min.js"></script>
<script src="js/bootstrap.js"></script>

</div>
</body>

</html>

funcionesEmpleado.views.php
<?php
require __DIR__ . "/partials/inmobiliaria.part.php";

ob_start();

?>

<div class="container col 12 m-5">
    <div class="row text-left">

        <form action="funcionesAdministrador.php" method="POST" enctype="multipart/form-data">
            <h1>Registra a tus empleados para que puedan vender, alquilar y borrar pisos.</h1>

            <div class="form-group">
                <label class="text-left" for="formGroupExampleInput">Usuario</label>
                <input type="text" class="form-control" id="nombre" name="usuario" placeholder="Ejemplo: TzarDaniel" required>
            </div>

            <div class="form-group">
                <label class="text-left" for="formGroupExampleInput">Nombre</label>
                <input type="text" class="form-control" id="nombre" onblur="comprobarLetrasNombre();" name="nombre" placeholder="Ejemplo: Daniel" required>
            </div>
            <div class="form-group">
                <label class="text-left" for="formGroupExampleInput2">Apellido</label>
                <input type="text" class="form-control" id="apellido" onblur="comprobarLetrasApellido();" name="apellido" placeholder="Ejemplo: Gerchev" required>
            </div>
            <div class="form-group">
                <label class="text-left" for="formGroupExampleInput2">Telefono</label>

```

```

<input type="text" class="form-control" id="telefono" onblur="comprobarTelefono();" name="telefono"
placeholder="Ejemplo:653366987" required>
</div>

<div class="form-group">
    <label class="text-left" for="exampleInputEmail1">Email</label>
    <input type="email" class="form-control" aria-describedby="emailHelp" name="correo"
placeholder="Ejemplo: divaylov@ieslavereda.es" required>
    <small class="form-text text-muted">Nunca vamos a dar su informacion a terceros.</small>
</div>
<div class="form-group">
    <label class="text-left" for="exampleInputPassword1">Contraseña</label>
    <input type="password" class="form-control" name="contra" placeholder="Introduce contraseña" required>
</div>

<input type="submit" value="Enviar">
</form>

</div>
</div>

<SCRIPT LANGUAGE="Javascript">
let arCampos=document.getElementsByTagName('input');

let regNumero=/^\d+$/;
let regLetras=/^D+$/;

function comprobarLetrasNombre(){
    if (regLetras.test(arCampos[1].value)) {
        arCampos[1].style.background="green";
    }else{
        arCampos[1].style.background="red";
    }
}

function comprobarLetrasApellido(){
    if (regLetras.test(arCampos[2].value)) {
        arCampos[2].style.background="green";
    }else{
        arCampos[2].style.background="red";
    }
}

function comprobarTelefono(){
    if (regNumero.test(arCampos[3].value)) {
        arCampos[3].style.background="green";
    }else{
        arCampos[3].style.background="red";
    }
}

</SCRIPT>

<!-- Footer -->
<footer class="py-1 bg-dark ">
    <div class="container">
        <div class="row">
            <div class="col-3">
                <p class="text-primary "> Daniel Ivaylov</p>

```

```

</div>

<div class="col-3">
    <a href="index.php"><button type="button" class="btn btn-primary">Inicio</button></a>
</div>

<div class="col-3">
    <p class="text-primary ">Inmobiliaria en: C/San Sebastian 5 Valencia CP 46019</p>
</div>

<div class="col-3">
    <marquee class="text-primary " behavior="scroll" direction="left"> Informacion al 961 616 161 </marquee>
</div>

</div>
</div>
</footer>


<script src="js/jquery-3.3.1.min.js"></script>
<script src="js/bootstrap.js"></script>

</div>
</body>

</html>

funcionesUsuario.views.php
<?php
require __DIR__ . "/partials/inmobiliaria.part.php";
ob_start();

require_once 'Connection.php';
$PDO=Connection::make();
$idPropietario=$_SESSION['id'];

$statement=$PDO->prepare("SELECT * FROM propietario where id=:idProp");
$statement->setFetchMode(PDO::FETCH_CLASS|PDO::FETCH_PROPS_LATE,"propietario");
$statement->bindParam(':idProp', $idPropietario);
$statement->execute();
$datosPropietario=$statement->fetchAll();
?>

<form action="funcionesUsuario.php" method="POST" enctype="multipart/form-data">
    <h1>Aqui puedes añadir y borrar tus pisos, <?php echo $datosPropietario[0]['nombre']; ?></h1>
    <p>Si alguien esta interesado en sus inmuebles nosotros nos pondremos en contacto con usted con los datos que nos ha facilitado, estos no seran publicos y siempre que alguien le llame sera un profesional de nuestra inmobiliaria:</p>

    <table class="table table-striped">
        <tbody>
            <tr>
                <th scope="row">Telefono:</th>
                <td><?php echo $datosPropietario[0]['telefono']; ?></td>
                <th>Correo:</th>
                <td><?php echo $datosPropietario[0]['correo']; ?></td>
            </tr>
        </tbody>
    </table>

```

```

</tbody>
</table>

<h1>Registrar nuevo inmueble</h1>

<div class="form-row">

<div class="col-lg-3 col-md-6 mb-4">
    <label for="validationTooltip01">Precio alquiler</label>
    <input type="text" name="alquiler" onblur="comprobarNumerosAlquiler();" placeholder="Entero" >
</div>

<div class="col-lg-3 col-md-6 mb-4">
    <label for="validationTooltip01">Precio venta</label>
    <input type="text" name="venta" onblur="comprobarNumerosVenta();" placeholder="Entero" >
</div>

<div class="col-lg-3 col-md-6 mb-4">
    <label for="validationTooltip01">Metros cuadrados</label>
    <input type="text" class="form-control" onblur="comprobarNumerosMetros();" name="metros" placeholder="Metros" required>
</div>

<div class="col-lg-3 col-md-6 mb-4">
    <label for="validationTooltip01">Calle</label>
    <input type="text" class="form-control" name="calle" onblur="comprobarLetrasCalle();" placeholder="Nombre de la calle" required>
</div>

<div class="col-lg-3 col-md-6 mb-4">
    <label for="validationTooltip01">Numero</label>
    <input type="text" class="form-control" name="numero" onblur="comprobarNumeroCalle();" placeholder="Entero" required>
</div>

<div class="col-lg-3 col-md-6 mb-4">
    <label for="validationTooltip01">Puerta</label>
    <input type="text" class="form-control" name="puerta" onblur="comprobarNumeroPuerta();" placeholder="Entero" required>
</div>

<div class="col-lg-3 col-md-6 mb-4">
    <label for="validationTooltip01">Ciudad</label>
    <input type="text" class="form-control" name="ciudad" onblur="comprobarLetrasCiudad();" placeholder="First name" required>
</div>

<div class="col-lg-3 col-md-6 mb-4">
    <label for="validationTooltip01">Telefono</label>
    <input type="text" class="form-control" name="telefonoPropietario" onblur="comprobarNumerosTelefono();" placeholder="Entero" required>
</div>

```

```

<div class="col-md-4 mb-3">
  <label class="my-1 mr-2" for="inlineFormCustomSelectPref">Tipo inmueble</label>
  <select class="custom-select my-1 mr-sm-2" name="tipo" id="habitaciones">
    <option selected value="chalet">Chalet</option>
    <option value="finca">Finca</option>
    <option value="piso">Piso</option>
    <option value="terreno">Terreno</option>
    <option value="garaje">Garaje</option>
    <option value="trastero">Trastero</option>
  </select>
</div>

<div class="col-md-4 mb-3">
  <label class="my-1 mr-2" for="inlineFormCustomSelectPref">Ascensor</label>
  <select class="custom-select my-1 mr-sm-2" name="ascensor" id="habitaciones">
    <option selected value=null>No</option>
    <option value="si">Si</option>
  </select>
</div>

<div class="col-md-4 mb-3">
  <label class="my-1 mr-2" for="inlineFormCustomSelectPref">Habitaciones</label>
  <select class="custom-select my-1 mr-sm-2" name="habitaciones" id="habitaciones">
    <option selected value="1">1</option>
    <option value="2">2</option>
    <option value="3">3</option>
    <option value="4">4</option>
    <option value="5">5</option>
    <option value="6">6</option>
    <option value="7">7</option>
    <option value="8">8</option>
    <option value="9">9</option>
    <option value="10">10</option>
  </select>
</div>

<div class="col-md-4 mb-3">
  <label class="my-1 mr-2" for="inlineFormCustomSelectPref">Piso</label>
  <select class="custom-select my-1 mr-sm-2" name="piso" id="habitaciones">
    <option selected value="1">1</option>
    <option value="2">2</option>
    <option value="3">3</option>
    <option value="4">4</option>
    <option value="5">5</option>
    <option value="6">6</option>
    <option value="7">7</option>
    <option value="8">8</option>
    <option value="9">9</option>
    <option value="11">11</option>
    <option value="12">12</option>
    <option value="13">13</option>
    <option value="14">14</option>
    <option value="15">15</option>
    <option value="16">16</option>
    <option value="17">17</option>
    <option value="18">18</option>
    <option value="19">19</option>
    <option value="20">20</option>
  </select>
</div>

```

```

</div>

<div class="col-md-4 mb-3">
  <label class="my-1 mr-2" for="inlineFormCustomSelectPref">Provincia</label>
  <select class="custom-select my-1 mr-sm-2" name="provincia" id="provincia">
    <option value='alava'>Álava</option>
    <option value='albacete'>Albacete</option>
    <option value='alicante'>Alicante/Alacant</option>
    <option value='almeria'>Almería</option>
    <option value='asturias'>Asturias</option>
    <option value='avila'>Ávila</option>
    <option value='badajoz'>Badajoz</option>
    <option value='barcelona'>Barcelona</option>
    <option value='burgos'>Burgos</option>
    <option value='caceres'>Cáceres</option>
    <option value='cadiz'>Cádiz</option>
    <option value='cantabria'>Cantabria</option>
    <option value='castellon'>Castellón/Castelló</option>
    <option value='ceuta'>Ceuta</option>
    <option value='ciudadreal'>Ciudad Real</option>
    <option value='cordoba'>Córdoba</option>
    <option value='cuenca'>Cuenca</option>
    <option value='girona'>Girona</option>
    <option value='laspalmas'>Las Palmas</option>
    <option value='granada'>Granada</option>
    <option value='guadalajara'>Guadalajara</option>
    <option value='guipuzcoa'>Guipúzcoa</option>
    <option value='huelva'>Huelva</option>
    <option value='huesca'>Huesca</option>
    <option value='illesbalears'>Illes Balears</option>
    <option value='jaen'>Jaén</option>
    <option value='acoruña'>A Coruña</option>
    <option value='larioja'>La Rioja</option>
    <option value='leon'>León</option>
    <option value='lleida'>Lleida</option>
    <option value='lugo'>Lugo</option>
    <option value='madrid'>Madrid</option>
    <option value='malaga'>Málaga</option>
    <option value='melilla'>Melilla</option>
    <option value='murcia'>Murcia</option>
    <option value='navarra'>Navarra</option>
    <option value='ourense'>Ourense</option>
    <option value='palencia'>Palencia</option>
    <option value='pontevedra'>Pontevedra</option>
    <option value='salamanca'>Salamanca</option>
    <option value='segovia'>Segovia</option>
    <option value='sevilla'>Sevilla</option>
    <option value='soria'>Soria</option>
    <option value='tarragona'>Tarragona</option>
    <option value='santacruztenerife'>Santa Cruz de Tenerife</option>
    <option value='teruel'>Teruel</option>
    <option value='toledo'>Toledo</option>
    <option value='valencia'>Valencia/Valéncia</option>
    <option value='valladolid'>Valladolid</option>
    <option value='vizcaya'>Vizcaya</option>
    <option value='zamora'>Zamora</option>
    <option value='zaragoza'>Zaragoza</option>
  </select>
</div>

```

```

<div class="col-md-4 mb-3">
    <label for="validationTooltip01">Codigo postal</label>
    <input type="text" class="form-control" onblur="comprobarNumerosCP();" name="cp" placeholder="CP"
required>
</div>

<div class="col-md-4 mb-3">
    <label for="num">Foto 1 </label>
    <input type="file" name="foto1" value="">
</div>

<div class="col-md-4 mb-3">
    <label for="num">Foto 2 </label>
    <input type="file" name="foto2" value="">
</div>

<div class="col-md-4 mb-3">
    <label for="num">Foto 3 </label>
    <input type="file" name="foto3" value="">
</div>

<div class="col-md-4 mb-3">
    <label for="num">Foto 4 </label>
    <input type="file" name="foto4" value="">
</div>

<div class="col-md-4 mb-3">
    <label for="num">Foto 5 </label>
    <input type="file" name="foto5" value="">
</div>

<div class="form-group col-12">
    <label for="coment">Descripcion del inmueble</label>
    <textarea class="form-control" id="descripcion" name="descripcion" placeholder="Describenos de la mejor
manera las caracteristicas del inmueble. Recuerda que de eso depende su venta/alquiler." rows="3"></textarea>
</div>

</div>
<div> <input type="submit" value="Enviar"></input>
</div>
<br>
</form>

<div class="container">

<div class="row justify-content-around">
    <div class="col">
        <a title="Borra tus inmuebles" href="usuarioBorraInmueble.php">
            
        </div>
    </div>
</div>

```

<!-- Expresiones regulares javascript -->

```

<SCRIPT LANGUAGE="Javascript">
let arCampos=document.getElementsByTagName('input');

let regNumero=/^\d+$/;
let regLetras=/^D+$/;

function comprobarNumerosAlquiler(){
    if (regNumero.test(arCampos[0].value)) {
        arCampos[0].style.background="green";
    }else{
        arCampos[0].style.background="red";
    }
}

function comprobarNumerosVenta(){
    if (regNumero.test(arCampos[1].value)) {
        arCampos[1].style.background="green";
    }else{
        arCampos[1].style.background="red";
    }
}

function comprobarNumerosMetros(){
    if (regNumero.test(arCampos[2].value)) {
        arCampos[2].style.background="green";
    }else{
        arCampos[2].style.background="red";
    }
}

function comprobarLetrasCalle(){
    if (regLetras.test(arCampos[3].value)) {
        arCampos[3].style.background="green";
    }else{
        arCampos[3].style.background="red";
    }
}

function comprobarNumeroCalle(){
    if (regNumero.test(arCampos[4].value)) {
        arCampos[4].style.background="green";
    }else{
        arCampos[4].style.background="red";
    }
}

function comprobarNumeroPuerta(){
    if (regNumero.test(arCampos[5].value)) {
        arCampos[5].style.background="green";
    }else{
        arCampos[5].style.background="red";
    }
}

function comprobarLetrasCiudad(){
    if (regLetras.test(arCampos[6].value)) {
        arCampos[6].style.background="green";
    }
}

```

```

        }else{
            arCampos[6].style.background="red";
        }
    }

function comprobarNumerosTelefono(){
    if (regNumero.test(arCampos[7].value)) {
        arCampos[7].style.background="green";
    }else{
        arCampos[7].style.background="red";
    }
}

function comprobarNumerosCP(){
    if (regNumero.test(arCampos[8].value)) {
        arCampos[8].style.background="green";
    }else{
        arCampos[8].style.background="red";
    }
}
</SCRIPT>

```

```

<!-- Footer -->
<footer class="py-1 bg-dark">
    <div class="container">
        <div class="row">
            <div class="col-3">
                <p class="text-primary"> Daniel Ivaylov</p>
            </div>

            <div class="col-3">
                <a href="index.php"><button type="button" class="btn btn-primary">Inicio</button></a>
            </div>

            <div class="col-3">
                <p class="text-primary">Inmobiliaria en: C/San Sebastian 5 Valencia CP 46019</p>
            </div>

            <div class="col-3">
                <marquee class="text-primary" behavior="scroll" direction="left"> Informacion al 961 616 161 </marquee>
            </div>
        </div>
    </div>
</footer>

```

```

<!-- Bootstrap JavaScript -->
<script src="js/jquery-3.3.1.min.js"></script>
<script src="js/bootstrap.js"></script>

</div>
</body>

</html>

```

```

imprimirAlquiler.views.php
<?php
ob_start();

require __DIR__."/partials/inmobiliaria.part.php";
require_once 'Connection.php';
$PDO=Connection::make();

$objetos=[];

if ($_SERVER['REQUEST_METHOD']=='POST') {
//SELECT * FROM inmueble where alquiler< 5000 and tipo = 'chalet' and ciudad = 'Valencia' and habitaciones< 4
$dinero=$_POST['dinero'];
$tipo=$_POST['tipo'];
$ciudad=$_POST['ciudad'];
$habitaciones=$_POST['habitaciones'];

$statement=$PDO->prepare("SELECT * FROM inmueble where alquiler<:dinero and alquiler>0 and tipo = :tipo and
ciudad = :ciudad and habitaciones>= :habitaciones");
$statement->setFetchMode(PDO::FETCH_CLASS|PDO::FETCH_PROPS_LATE,"inmueble");
$statement->bindParam(':dinero', $dinero);
$statement->bindParam(':tipo', $tipo);
$statement->bindParam(':ciudad', $ciudad);
$statement->bindParam(':habitaciones', $habitaciones);
$statement->execute();
$objetos=$statement->fetchAll();
// print_r($objetos);

if (!$objetos) {
//no hay objetos redirigimos atras
header('Location:alquilar.php');

}

}

?>

<div class="container m-2">

<div class="row text-center ">
<?php foreach($objetos as $objeto): ?>
<div class="col-lg-3 col-md-6 mb-4">

<div class="card h-100">
<img class="card-img-top circulito zoomGrande" src=<?php echo $objeto['foto1'];?>>
<div class="card-body">
<h4 class="card-title">Ciudad <?php echo $objeto['ciudad'];?> </h4>
<p class="card-text">
    Informacion sobre inmueble:
</p>

<p class="text-left card-text">
```

```

        Tipo de inmueble: <?php echo $objeto['tipo'];?></br>
        Alquiler: <?php echo $objeto['alquiler'];?></br>
        Metros cuadrados: <?php echo $objeto['metros'];?></br>
        Habitaciones: <?php echo $objeto['habitaciones'];?></br>
        Altura: <?php echo $objeto['piso'];?></br>
        Numero de referencia:
        </p>
        <form action="masInformacion.php" method="POST" enctype="multipart/form-data">
            <input type="text" readonly name="id" value=<?php echo $objeto['id'];?> />

        </div>
        <div class="card-footer">
            <!--class="btn btn-primary submit"-->
            <input class="btn btn-primary" type="submit" value="Mas informacion"></input>
        </div>
        </form>

    </div>
</div>

<?php endforeach; ?>

</div>

</div>

<div class="container m-2">

    <div class="row justify-content-around contraste">
        <a title="Volver atras" href="alquilar.php">
            
        </div>
    </div>

<!-- Footer -->
<footer class="py-1 bg-dark ">
    <div class="container">
        <div class="row">
            <div class="col-3">
                <p class="text-primary "> Daniel Ivaylov</p>
            </div>

            <div class="col-3">
                <a href="index.php"><button type="button" class="btn btn-primary">Inicio</button></a>
            </div>

            <div class="col-3">
                <p class="text-primary ">Inmobiliaria en: C/San Sebastian 5 Valencia CP 46019</p>
            </div>

            <div class="col-3">
                <marquee class="text-primary " behavior="scroll" direction="left"> Informacion al 961 616 161 </marquee>
            </div>
        </div>
    </div>
</footer>

```

```

<!-- Bootstrap JavaScript -->
<script src="js/jquery-3.3.1.min.js"></script>
<script src="js/bootstrap.js"></script>

</div>
</body>

</html>
imprimirComprar.views.php
<?php
ob_start();

require __DIR__."/partials/inmobiliaria.part.php";
require_once 'Connection.php';
$PDO=Connection::make();

$objetos=[];

if ($_SERVER['REQUEST_METHOD']=='POST') {
//SELECT * FROM inmueble where alquiler< 5000 and tipo = 'chalet' and ciudad = 'Valencia' and habitaciones< 4
$dinero=$_POST['dinero'];
$tipo=$_POST['tipo'];
$ciudad=$_POST['ciudad'];
$habitaciones=$_POST['habitaciones'];

$stmt=$PDO->prepare("SELECT * FROM inmueble where venta<:dinero and venta>0 and tipo = :tipo and ciudad
=:ciudad and habitaciones>= :habitaciones");
$stmt->setFetchMode(PDO::FETCH_CLASS|PDO::FETCH_PROPS_LATE,"inmueble");
$stmt->bindParam(':dinero', $dinero);
$stmt->bindParam(':tipo', $tipo);
$stmt->bindParam(':ciudad', $ciudad);
$stmt->bindParam(":habitaciones", $habitaciones);
$stmt->execute();
$objetos=$stmt->fetchAll();
// print_r($objetos);

if (!$objetos) {
    //no hay objetos redirigimos atras

    header('Location:comprar.php');

}
}

?>

<div class="container m-2">
<div class="row text-center">

```

```

<?php foreach($objetos as $objeto): ?>
<div class="col-lg-3 col-md-6 mb-4">

    <div class="card h-100">
        <img class="card-img-top circulito zoomGrande" src=<?php echo $objeto['foto1'];?>>
        <div class="card-body">
            <h4 class="card-title">Ciudad <?php echo $objeto['ciudad'];?> </h4>
            <p class="card-text">
                Informacion sobre inmueble:
            </p>

            <p class="text-left card-text">
                Tipo de inmueble: <?php echo $objeto['tipo'];?><br>
                Precio: <?php echo $objeto['venta'];?><br>
                Metros cuadrados: <?php echo $objeto['metros'];?><br>
                Habitaciones: <?php echo $objeto['habitaciones'];?><br>
                Altura: <?php echo $objeto['piso'];?><br>
                Numero de referencia:
            </p>
            <form action="masInformacion.php" method="POST" enctype="multipart/form-data">
                <input type="text" readonly name="id" value=<?php echo $objeto['id'];?> />

            </div>
            <div class="card-footer">
                <!--class="btn btn-primary submit"-->
                <input class="btn btn-primary" type="submit" value="Mas informacion"></input>
            </div>
            </form>

        </div>
    </div>
</div>

<?php endforeach; ?>

</div>

</div>

<div class="container m-2">

    <div class="row justify-content-around contraste">
        <a title="Volver atras" href="comprar.php">
            
        </div>
    </div>

<!-- Footer -->
<footer class="py-1 bg-dark ">
    <div class="container">
        <div class="row">
            <div class="col-3">
                <p class="text-primary "> Daniel Ivaylov</p>
            </div>

            <div class="col-3">
                <a href="index.php"><button type="button" class="btn btn-primary">Inicio</button></a>
            </div>
        </div>
    </div>

```

```

<div class="col-3">
    <p class="text-primary ">Inmobiliaria en: C/San Sebastian 5 Valencia CP 46019</p>
</div>

<div class="col-3">
    <marquee class="text-primary " behavior="scroll" direction="left"> Informacion al 961 616 161 </marquee>
</div>

</div>
</div>
</footer>

<!-- Bootstrap JavaScript -->
<script src="js/jquery-3.3.1.min.js"></script>
<script src="js/bootstrap.js"></script>

</div>
</body>

</html>

index.views.php
<?php
ob_start();

require __DIR__ . "/partials/inmobiliaria.part.php";
require_once 'Connection.php';
$PDO=Connection::make();

$objetos=[];

if ($_SERVER['REQUEST_METHOD']=='POST') {
//SELECT * FROM inmueble where alquiler< 5000 and tipo = 'chalet' and ciudad = 'Valencia' and habitaciones< 4
$dinero=$_POST['dinero'];
$tipo=$_POST['tipo'];
$ciudad=$_POST['ciudad'];
$habitaciones=$_POST['habitaciones'];

$statement=$PDO->prepare("SELECT * FROM inmueble where venta<:dinero and venta>0 and tipo = :tipo and ciudad = :ciudad and habitaciones>= :habitaciones");
$statement->setFetchMode(PDO::FETCH_CLASS|PDO::FETCH_PROPS_LATE,"inmueble");
$statement->bindParam(':dinero', $dinero);
$statement->bindParam(':tipo', $tipo);
$statement->bindParam(':ciudad', $ciudad);
$statement->bindParam(':habitaciones', $habitaciones);
$statement->execute();
$objetos=$statement->fetchAll();
// print_r($objetos);

if (!$objetos) {
    //no hay objetos redirigimos atras

    header('Location:comprar.php');
}
}

```

```

    }

}

?>

<div class="container m-2">

    <div class="row text-center">
        <?php foreach($objetos as $objeto): ?>
            <div class="col-lg-3 col-md-6 mb-4">

                <div class="card h-100">
                    <img class="card-img-top circulito zoomGrande" src=<?php echo $objeto['foto1'];?>>
                    <div class="card-body">
                        <h4 class="card-title">Ciudad <?php echo $objeto['ciudad'];?> </h4>
                        <p class="card-text">
                            Informacion sobre inmueble:
                        </p>

                        <p class="text-left card-text">
                            Tipo de inmueble: <?php echo $objeto['tipo'];?><br>
                            Precio: <?php echo $objeto['venta'];?><br>
                            Metros cuadrados: <?php echo $objeto['metros'];?><br>
                            Habitaciones: <?php echo $objeto['habitaciones'];?><br>
                            Altura: <?php echo $objeto['piso'];?><br>
                            Numero de referencia:
                        </p>
                        <form action="masInformacion.php" method="POST" enctype="multipart/form-data">
                            <input type="text" readonly name="id" value=<?php echo $objeto['id'];?> />

                        </div>
                        <div class="card-footer">
                            <!--class="btn btn-primary submit"-->
                            <input class="btn btn-primary" type="submit" value="Mas informacion"></input>
                        </div>
                    </form>

                </div>
            </div>

        <?php endforeach; ?>

    </div>

    </div>

<div class="container m-2">

    <div class="row justify-content-around contraste">
        <a title="Volver atras" href="comprar.php">
            
        </div>
    </div>

```

```

<!-- Footer -->
<footer class="py-1 bg-dark ">
  <div class="container">
    <div class="row">
      <div class="col-3">
        <p class="text-primary "> Daniel Ivaylov</p>
      </div>

      <div class="col-3">
        <a href="index.php"><button type="button" class="btn btn-primary">Inicio</button></a>
      </div>

      <div class="col-3">
        <p class="text-primary ">Inmobiliaria en: C/San Sebastian 5 Valencia CP 46019</p>
      </div>

      <div class="col-3">
        <marquee class="text-primary " behavior="scroll" direction="left"> Informacion al 961 616 161 </marquee>
      </div>

    </div>
  </div>
</footer>

```

```

<!-- Bootstrap JavaScript -->
<script src="js/jquery-3.3.1.min.js"></script>
<script src="js/bootstrap.js"></script>

</div>
</body>

</html>

```

```

index.views.php
<?php
ob_start();

require __DIR__."/partials/inmobiliaria.part.php";
require_once 'Connection.php';
$PDO=Connection::make();

$objetos=[];

if ($_SERVER['REQUEST_METHOD']=='GET') {

  $stmt=$PDO->prepare("SELECT * FROM inmueble");
  $stmt->setFetchMode(PDO::FETCH_CLASS|PDO::FETCH_PROPS_LATE,"inmueble");
  $stmt->execute();
  $objetos=$stmt->fetchAll();
}

?>

```

```

<div class="container">

    <!-- Header -->
    <header class="jumbotron my-4">
        <h1 class="display-3">Queremos encontrar tu hogar</h1>
        <p>Que nos diferencia? En nuestra empresa no existe la palabra exclusividad!</p>
        <p>La mayoria de inmobiliarias les obliga a tener su inmueble durante meses por contrato pero nosotros creemos en nuestros agentes y como grandes profesionales que son les garantizamos la venta de su inmueble. Si quiere usted mismo puede dar de baja su piso! Sin ningun compromismo.</p>
        <p>Les ofrecemos asesoramiento totalmente gratuito. Tasacion de su propiedad. Compra venta y alquiler de toda clase de inmuebles.</p>
        <a href="contacto.php" class="btn btn-primary btn-lg">Quiero saber mas!</a>
    </header>

    <h1 class="text-center ">A continuacion nuestra mejor seleccion de inmuebles</h1>
    <div class="row text-center">
        <?php foreach($objetos as $objeto): ?>
        <div class="col-lg-3 col-md-6">

            <div class="card h-100">
                <img class="card-img-top zoom" src=<?php echo $objeto['foto1'];?>>
                <div class="card-body">
                    <h4 class="card-title">Ciudad <?php echo $objeto['ciudad'];?> </h4>
                    <p class="card-text">
                        Informacion sobre inmueble:
                    </p>

                    <p class="text-left card-text">
                        Tipo de inmueble: <?php echo $objeto['tipo'];?><br>
                        Habitaciones: <?php echo $objeto['habitaciones'];?><br>
                        Metros cuadrados: <?php echo $objeto['metros'];?><br>
                        Altura: <?php echo $objeto['piso'];?><br>
                        Numero de referencia:
                    </p>
                    <form action="masInformacion.php" method="POST" enctype="multipart/form-data">
                        <input type="text" readonly name="id" value=<?php echo $objeto['id'];?> />
                    </div>
                    <div class="card-footer">
                        <!--class="btn btn-primary submit"-->
                        <input class="btn btn-primary" type="submit" value="Mas informacion"></input>
                    </div>
                    </form>
                </div>
            </div>
        <?php endforeach; ?>

    </div>
    </div>

    <!-- Footer -->
    <footer class="py-1 bg-dark ">

```

```

<div class="container">
  <div class="row">
    <div class="col-3">
      <p class="text-primary "> Daniel Ivaylov</p>
    </div>

    <div class="col-3">
      <a href="index.php"><button type="button" class="btn btn-primary">Inicio</button></a>
    </div>

    <div class="col-3">
      <p class="text-primary ">Inmobiliaria en: C/San Sebastian 5 Valencia CP 46019</p>
    </div>

    <div class="col-3">
      <marquee class="text-primary " behavior="scroll" direction="left"> Informacion al 961 616 161 </marquee>
    </div>

  </div>
</div>
</footer>

<!-- Bootstrap JavaScript -->
<script src="js/jquery-3.3.1.min.js"></script>
<script src="js/bootstrap.js"></script>

</div>
</body>

</html>
loguearAdministrador.views.php

<?php
ob_start();

require __DIR__ . "/partials/inmobiliaria.part.php";

?>

<div class="container m-2">
  <div class="row text-left">

    <form action="loguearAdministrador.php" method="POST" enctype="multipart/form-data">
      <h1>Seccion de administrador, introduce tus datos a continuacion para dar de alta nuevos empleados.</h1>

      <div class="form-group">
        <label for="formGroupExampleInput">Usuario</label>
        <input type="text" class="form-control" id="nombre" name="usuario" placeholder="Ejemplo: TzarDaniel">
      </div>
      <div class="form-group">
        <label for="exampleInputPassword1">Contraseña</label>
        <input type="password" class="form-control" name="contra" placeholder="Introduce contraseña">
      </div>

      <input type="submit" value="Enviar"></input>
    </form>
  </div>
</div>

```

```

</div>
</div>

<!-- Footer -->
<footer class="py-1 bg-dark">
  <div class="container">
    <div class="row">
      <div class="col-3">
        <p class="text-primary"> Daniel Ivaylov</p>
      </div>

      <div class="col-3">
        <a href="index.php"><button type="button" class="btn btn-primary">Inicio</button></a>
      </div>

      <div class="col-3">
        <p class="text-primary">Inmobiliaria en: C/San Sebastian 5 Valencia CP 46019</p>
      </div>

      <div class="col-3">
        <marquee class="text-primary" behavior="scroll" direction="left"> Informacion al 961 616 161 </marquee>
      </div>
    </div>
  </div>
</footer>

```

```

<!-- Bootstrap JavaScript -->
<script src="js/jquery-3.3.1.min.js"></script>
<script src="js/bootstrap.js"></script>

</div>
</body>

</html>
loguearEmpleado.views.php
<?php
ob_start();

require __DIR__ . "/partials/inmobiliaria.part.php";

?>

<div class="container m-2">
  <div class="row text-left">

    <form action="loguearEmpleado.php" method="POST" enctype="multipart/form-data">
      <h1>Seccion de empleados. Ingresa tus datos para vender y alquilar inmuebles.</h1>

      <div class="form-group">
        <label for="formGroupExampleInput">Usuario</label>
        <input type="text" class="form-control" id="nombre" name="usuario" placeholder="Ejemplo: TzarDaniel">
      </div>
      <div class="form-group">

```

```

<label for="exampleInputPassword1">Contraseña</label>
<input type="password" class="form-control" name="contra" placeholder="Introduce contraseña">
</div>

<input type="submit" value="Enviar"></input>
</form>

</div>
</div>

<!-- Footer -->
<footer class="py-1 bg-dark ">
<div class="container">
<div class="row">
<div class="col-3">
<p class="text-primary "> Daniel Ivaylov</p>
</div>

<div class="col-3">
<a href="index.php"><button type="button" class="btn btn-primary">Inicio</button></a>
</div>

<div class="col-3">
<p class="text-primary ">Inmobiliaria en: C/San Sebastian 5 Valencia CP 46019</p>
</div>

<div class="col-3">
<marquee class="text-primary " behavior="scroll" direction="left"> Informacion al 961 616 161 </marquee>
</div>

</div>
</div>
</footer>
```

```

<!-- Bootstrap JavaScript -->
<script src="js/jquery-3.3.1.min.js"></script>
<script src="js/bootstrap.js"></script>

</div>
</body>

</html>
loguearUsuario.views.php
<?php
ob_start();

require __DIR__ . "/partials/inmobiliaria.part.php";

?>

<div class="container m-2">
<div class="row text-left">
<form action="loguearUsuario.php" method="POST" enctype="multipart/form-data">
<h1>Si te has registrado puedes loguear para gestionar tus inmuebles</h1>
```

```

<div class="form-group col-12">
    <label for="formGroupExampleInput">Usuario</label>
    <input type="text" class="form-control" id="nombre" name="usuario" placeholder="Ejemplo: TzarDaniel">
</div>
<div class="form-group col-12">
    <label for="exampleInputPassword1">Contraseña</label>
    <input type="password" class="form-control" name="contra" placeholder="Introduce contraseña">
</div>

<input type="submit" value="Enviar"></input>
</form>

</div>
</div>

<!-- Footer -->
<footer class="py-1 bg-dark ">
    <div class="container">
        <div class="row">
            <div class="col-3">
                <p class="text-primary "> Daniel Ivaylov</p>
            </div>

            <div class="col-3">
                <a href="index.php"><button type="button" class="btn btn-primary">Inicio</button></a>
            </div>

            <div class="col-3">
                <p class="text-primary ">Inmobiliaria en: C/San Sebastian 5 Valencia CP 46019</p>
            </div>

            <div class="col-3">
                <marquee class="text-primary " behavior="scroll" direction="left"> Informacion al 961 616 161 </marquee>
            </div>
        </div>
    </div>
</footer>

<!-- Bootstrap JavaScript -->
<script src="js/jquery-3.3.1.min.js"></script>
<script src="js/bootstrap.js"></script>

</div>
</body>

</html>

masInformacion.views.php
<?php
ob_start();

require __DIR__ . "/partials/inmobiliaria.part.php";

```

```

require_once 'Connection.php';
$PDO=Connection::make();

if ($_SERVER['REQUEST_METHOD']=='POST') {

    $sql = "SELECT * from inmueble where id=:id";
    $id=$_POST['id'];

    $statement=$PDO->prepare($sql);
    $statement->bindParam(':id', $id);
    $statement->execute();
    $inmueble = $statement->fetch(PDO::FETCH_ASSOC);

    if (!$inmueble){
        ?><script>
            alert("No se ha encontrado inmueble con el id indicado.");
        </script>
        <?php
    }

    }else{

        $arImg=[];
        for ($i=1;$i<6;$i++){
            $nombre="foto".$i;
            if (null!=$inmueble[$nombre]){
                array_push($arImg,$inmueble[$nombre]);
            }
        }
    }

    ?>

<!-- Imprimir imagenes -->

<div id="myCarousel" class="carousel slide bg-inverse w-50 ml-auto mr-auto" data-ride="carousel">
    <div class="carousel-inner" role="listbox">

        <div class="carousel-item active zoomMuyGrande">
            <img class="d-block w-100" src=<?php echo $arImg[0];?> >
            <div class="carousel-caption">
            </div>
        </div>

        <?php foreach($arImg as $img): ?>
        <div class="carousel-item zoomMuyGrande">
            <img class="d-block w-100" src=<?php echo $img;?> >
            <div class="carousel-caption">
            </div>
        </div>
        <?php endforeach; ?>
    </div>
</div>

```

```

</div>

<a class="carousel-control-prev" href="#myCarousel" role="button" data-slide="prev">
  <span class="carousel-control-prev-icon" aria-hidden="true"></span>
  <span class="sr-only">Anterior</span>
</a>
<a class="carousel-control-next" href="#myCarousel" role="button" data-slide="next">
  <span class="carousel-control-next-icon" aria-hidden="true"></span>
  <span class="sr-only">Siguiente</span>
</a>
</div>

```

<!-- Datos a imprimir -->

```


| Referencia:   | <?php echo \$inmueble['id'];?>                                                                      |
|---------------|-----------------------------------------------------------------------------------------------------|
| Tipo:         | <?php echo \$inmueble['tipo'];?>                                                                    |
| Alquiler:     | <?php echo \$inmueble['alquiler'];?>                                                                |
| Venta:        | <?php echo \$inmueble['venta'];?>                                                                   |
| Habitaciones: | <?php echo \$inmueble['habitaciones'];?>                                                            |
| Metros:       | <?php echo \$inmueble['metros'];?>                                                                  |
| Ciudad:       | <?php echo \$inmueble['ciudad'];?>                                                                  |
| Provincia:    | <?php echo \$inmueble['provincia'];?>                                                               |
| Calle:        | <?php echo \$inmueble['calle'];?>                                                                   |
| Numero:       | <?php echo \$inmueble['numero'];?>                                                                  |
| Altura:       | <?php echo \$inmueble['piso'];?>                                                                    |
| Ascensor:     | <?php if('nu'!=\$inmueble['ascensor']){   echo \$inmueble['ascensor']; } else{   echo "No"; }></td> |


```

```

</tr>

<tr class="fila">
    <th scope="row">CP:</th>
    <td><?php echo $inmueble['cp'];?></td>
    <th scope="row"></th>
    <td></td>
</tr>

</tbody>
</table>

<table class="table table-striped fila">
    <tbody>
        <th scope="row">Comentario:</th>
        <td><?php echo $inmueble['descripcion'];?></td>
    </tbody>
</table>
<!-- Footer -->
<footer class="py-1 bg-dark ">
    <div class="container">
        <div class="row">
            <div class="col-3">
                <p class="text-primary "> Daniel Ivaylov</p>
            </div>

            <div class="col-3">
                <a href="index.php"><button type="button" class="btn btn-primary">Inicio</button></a>
            </div>

            <div class="col-3">
                <p class="text-primary ">Inmobiliaria en: C/San Sebastian 5 Valencia CP 46019</p>
            </div>

            <div class="col-3">
                <marquee class="text-primary " behavior="scroll" direction="left"> Informacion al 961 616 161 </marquee>
            </div>
        </div>
    </div>
</footer>

<!-- Bootstrap JavaScript -->
<script src="js/jquery-3.3.1.min.js"></script>
<script src="js/bootstrap.js"></script>

</div>
</body>

</html>
masInformacionEmpleado.views.php
<?php
ob_start();
```

```

require __DIR__ . "/partials/inmobiliaria.part.php";

require_once 'Connection.php';
$PDO=Connection::make();

if ($_SERVER['REQUEST_METHOD']=='POST') {

    $sql = "SELECT * from inmueble where id=:id";
    $id=$_POST['id'];

    $statement=$PDO->prepare($sql);
    $statement->bindParam(":id", $id);
    $statement->execute();
    $inmueble = $statement->fetch(PDO::FETCH_ASSOC);

    if (!$inmueble){
        ?><script>
            alert("No se ha encontrado inmueble con el id indicado.");
        </script>
        <?php
    }

    }else{

        $arImg=[];
        for ($i=1;$i<6;$i++){
            $nombre="foto".$i;
            if (null!=$inmueble[$nombre]){
                array_push($arImg,$inmueble[$nombre]);
            }
        }
    }

    ?>

<!-- Imprimir imagenes -->

<div id="myCarousel" class="carousel slide bg-inverse w-50 ml-auto mr-auto" data-ride="carousel">
    <div class="carousel-inner" role="listbox">

        <div class="carousel-item active zoomMuyGrande">
            <img class="d-block w-100" src=<?php echo $arImg[0];?> >
            <div class="carousel-caption">
                </div>
        </div>

        <?php foreach($arImg as $img): ?>
        <div class="carousel-item zoomMuyGrande">
            <img class="d-block w-100" src=<?php echo $img;?> >
            <div class="carousel-caption">
                </div>
        </div>
    </div>
</div>

```

```

<?php endforeach; ?>

</div>

<a class="carousel-control-prev" href="#myCarousel" role="button" data-slide="prev">
  <span class="carousel-control-prev-icon" aria-hidden="true"></span>
  <span class="sr-only">Anterior</span>
</a>
<a class="carousel-control-next" href="#myCarousel" role="button" data-slide="next">
  <span class="carousel-control-next-icon" aria-hidden="true"></span>
  <span class="sr-only">Siguiente</span>
</a>
</div>

<!-- Datos a imprimir --&gt;

&lt;table class="table table-striped"&gt;
&lt;tbody&gt;
&lt;tr class="fila"&gt;
  &lt;th scope="row"&gt;Referencia:&lt;/th&gt;
  &lt;td&gt;&lt;?php echo $inmueble['id'];?&gt;&lt;/td&gt;
  &lt;th&gt;Tipo:&lt;/th&gt;
  &lt;td&gt;&lt;?php echo $inmueble['tipo'];?&gt;&lt;/td&gt;
&lt;/tr&gt;
&lt;tr class="fila"&gt;
  &lt;th scope="row"&gt;Alquiler:&lt;/th&gt;
  &lt;td&gt;&lt;?php echo $inmueble['alquiler'];?&gt;&lt;/td&gt;
  &lt;th&gt;Venta:&lt;/th&gt;
  &lt;td&gt;&lt;?php echo $inmueble['venta'];?&gt;&lt;/td&gt;
&lt;/tr&gt;
&lt;tr class="fila"&gt;
  &lt;th scope="row"&gt;Habitaciones:&lt;/th&gt;
  &lt;td&gt;&lt;?php echo $inmueble['habitaciones'];?&gt;&lt;/td&gt;
  &lt;th&gt;Metros:&lt;/th&gt;
  &lt;td&gt;&lt;?php echo $inmueble['metros'];?&gt;&lt;/td&gt;
&lt;/tr&gt;
&lt;tr class="fila"&gt;
  &lt;th scope="row"&gt;Ciudad:&lt;/th&gt;
  &lt;td&gt;&lt;?php echo $inmueble['ciudad'];?&gt;&lt;/td&gt;
  &lt;th&gt;Provincia:&lt;/th&gt;
  &lt;td&gt;&lt;?php echo $inmueble['provincia'];?&gt;&lt;/td&gt;
&lt;/tr&gt;
&lt;tr class="fila"&gt;
  &lt;th scope="row"&gt;Calle:&lt;/th&gt;
  &lt;td&gt;&lt;?php echo $inmueble['calle'];?&gt;&lt;/td&gt;
  &lt;th&gt;Numero:&lt;/th&gt;
  &lt;td&gt;&lt;?php echo $inmueble['numero'];?&gt;&lt;/td&gt;
&lt;/tr&gt;
&lt;tr class="fila"&gt;
  &lt;th scope="row"&gt;Altura:&lt;/th&gt;
  &lt;td&gt;&lt;?php echo $inmueble['piso'];?&gt;&lt;/td&gt;
  &lt;th&gt;Ascensor:&lt;/th&gt;
  &lt;td&gt;&lt;?php if('nu'!=$inmueble['ascensor']){
    echo $inmueble['ascensor'];
  }else{
    echo "No";
  }?&gt;&lt;/td&gt;
&lt;/tr&gt;
</pre>

```

```

        }?></td>
    </tr>

<tr class="fila">
    <th scope="row">CP:</th>
    <td><?php echo $inmueble['cp'];?></td>
    <th scope="row">Telefono</th>
    <td><?php echo $inmueble['telefonoPropietario'];?></td>
</tr>

</tbody>
</table>

<table class="table table-striped fila">
    <tbody>
        <th scope="row">Comentario:</th>
        <td><?php echo $inmueble['descripcion'];?></td>
    </tbody>
</table>
<!-- Footer -->
<footer class="py-1 bg-dark ">
    <div class="container">
        <div class="row">
            <div class="col-3">
                <p class="text-primary "> Daniel Ivaylov</p>
            </div>

            <div class="col-3">
                <a href="index.php"><button type="button" class="btn btn-primary">Inicio</button></a>
            </div>

            <div class="col-3">
                <p class="text-primary ">Inmobiliaria en: C/San Sebastian 5 Valencia CP 46019</p>
            </div>

            <div class="col-3">
                <marquee class="text-primary " behavior="scroll" direction="left"> Informacion al 961 616 161 </marquee>
            </div>
        </div>
    </div>
</footer>

<!-- Bootstrap JavaScript -->
<script src="js/jquery-3.3.1.min.js"></script>
<script src="js/bootstrap.js"></script>

</div>
</body>

</html>

registrarAdministrador.views.php
<?php
```

```

ob_start();

require __DIR__ . "/partials/inmobiliaria.part.php";

require_once 'Connection.php';
$PDO=Connection::make();

if ($_SERVER['REQUEST_METHOD']=='POST') {

    $sql = "SELECT * from inmueble where id=:id";
    $id=$_POST['id'];

    $statement=$PDO->prepare($sql);
    $statement->bindParam(':id', $id);
    $statement->execute();
    $inmueble = $statement->fetch(PDO::FETCH_ASSOC);

    if (!$inmueble){
        ?><script>
            alert("No se ha encontrado inmueble con el id indicado.");
        </script>
        <?php
    }

    $arImg=[];
    for ($i=1;$i<6;$i++){
        $nombre="foto".$i;
        if (null!=$inmueble[$nombre]){
            array_push($arImg,$inmueble[$nombre]);
        }
    }
}

?>

<!-- Imprimir imagenes -->

<div id="myCarousel" class="carousel slide bg-inverse w-50 ml-auto mr-auto" data-ride="carousel">
<div class="carousel-inner" role="listbox">

    <div class="carousel-item active zoomMuyGrande">
        <img class="d-block w-100" src=<?php echo $arImg[0];?> >
        <div class="carousel-caption">
        </div>
    </div>

    <?php foreach($arImg as $img): ?>
    <div class="carousel-item zoomMuyGrande">
        <img class="d-block w-100" src=<?php echo $img;?> >
        <div class="carousel-caption">

```

```

        </div>
    </div>
    <?php endforeach; ?>

</div>

<a class="carousel-control-prev" href="#myCarousel" role="button" data-slide="prev">
    <span class="carousel-control-prev-icon" aria-hidden="true"></span>
    <span class="sr-only">Anterior</span>
</a>
<a class="carousel-control-next" href="#myCarousel" role="button" data-slide="next">
    <span class="carousel-control-next-icon" aria-hidden="true"></span>
    <span class="sr-only">Siguiente</span>
</a>
</div>

<!-- Datos a imprimir --&gt;

&lt;table class="table table-striped"&gt;
    &lt;tbody&gt;
        &lt;tr class="fila"&gt;
            &lt;th scope="row"&gt;Referencia:&lt;/th&gt;
            &lt;td&gt;&lt;?php echo $inmueble['id'];?&gt;&lt;/td&gt;
            &lt;th&gt;Tipo:&lt;/th&gt;
            &lt;td&gt;&lt;?php echo $inmueble['tipo'];?&gt;&lt;/td&gt;
        &lt;/tr&gt;
        &lt;tr class="fila"&gt;
            &lt;th scope="row"&gt;Alquiler:&lt;/th&gt;
            &lt;td&gt;&lt;?php echo $inmueble['alquiler'];?&gt;&lt;/td&gt;
            &lt;th&gt;Venta:&lt;/th&gt;
            &lt;td&gt;&lt;?php echo $inmueble['venta'];?&gt;&lt;/td&gt;
        &lt;/tr&gt;
        &lt;tr class="fila"&gt;
            &lt;th scope="row"&gt;Habitaciones:&lt;/th&gt;
            &lt;td&gt;&lt;?php echo $inmueble['habitaciones'];?&gt;&lt;/td&gt;
            &lt;th&gt;Metros:&lt;/th&gt;
            &lt;td&gt;&lt;?php echo $inmueble['metros'];?&gt;&lt;/td&gt;
        &lt;/tr&gt;
        &lt;tr class="fila"&gt;
            &lt;th scope="row"&gt;Ciudad:&lt;/th&gt;
            &lt;td&gt;&lt;?php echo $inmueble['ciudad'];?&gt;&lt;/td&gt;
            &lt;th&gt;Provincia:&lt;/th&gt;
            &lt;td&gt;&lt;?php echo $inmueble['provincia'];?&gt;&lt;/td&gt;
        &lt;/tr&gt;
        &lt;tr class="fila"&gt;
            &lt;th scope="row"&gt;Calle:&lt;/th&gt;
            &lt;td&gt;&lt;?php echo $inmueble['calle'];?&gt;&lt;/td&gt;
            &lt;th&gt;Numero:&lt;/th&gt;
            &lt;td&gt;&lt;?php echo $inmueble['numero'];?&gt;&lt;/td&gt;
        &lt;/tr&gt;
        &lt;tr class="fila"&gt;
            &lt;th scope="row"&gt;Altura:&lt;/th&gt;
            &lt;td&gt;&lt;?php echo $inmueble['piso'];?&gt;&lt;/td&gt;
            &lt;th&gt;Ascensor:&lt;/th&gt;
            &lt;td&gt;&lt;?php if('nu'!=$inmueble['ascensor']){
                echo $inmueble['ascensor'];
            } ?&gt;&lt;/td&gt;
        &lt;/tr&gt;
    &lt;/tbody&gt;
&lt;/table&gt;
</pre>

```

```

        }else{
            echo "No";
        }?></td>
    </tr>

<tr class="fila">
    <th scope="row">CP:</th>
    <td><?php echo $inmueble['cp'];?></td>
    <th scope="row">Telefono</th>
    <td><?php echo $inmueble['telefonoPropietario'];?></td>
</tr>

</tbody>
</table>

<table class="table table-striped fila">
    <tbody>
        <th scope="row">Comentario:</th>
        <td><?php echo $inmueble['descripcion'];?></td>
    </tbody>
</table>
<!-- Footer -->
<footer class="py-1 bg-dark ">
    <div class="container">
        <div class="row">
            <div class="col-3">
                <p class="text-primary "> Daniel Ivaylov</p>
            </div>

            <div class="col-3">
                <a href="index.php"><button type="button" class="btn btn-primary">Inicio</button></a>
            </div>

            <div class="col-3">
                <p class="text-primary ">Inmobiliaria en: C/San Sebastian 5 Valencia CP 46019</p>
            </div>

            <div class="col-3">
                <marquee class="text-primary " behavior="scroll" direction="left"> Informacion al 961 616 161 </marquee>
            </div>
        </div>
    </div>
</footer>

<!-- Bootstrap JavaScript -->
<script src="js/jquery-3.3.1.min.js"></script>
<script src="js/bootstrap.js"></script>

</div>
</body>

</html>

```

```

masInformacionEmpleado.views.php
<?php
ob_start();

require __DIR__ . "/partials/inmobiliaria.part.php";

require_once 'Connection.php';
$PDO=Connection::make();

if ($_SERVER['REQUEST_METHOD']=='POST') {

    $sql = "SELECT * from inmueble where id=:id";
    $id=$_POST['id'];

    $statement=$PDO->prepare($sql);
    $statement->bindParam(':id', $id);
    $statement->execute();
    $inmueble = $statement->fetch(PDO::FETCH_ASSOC);

    if (!$inmueble){
        ?><script>
            alert("No se ha encontrado inmueble con el id indicado.");
        </script>
        <?php
    }

    }else{

        $arImg=[];
        for ($i=1;$i<6;$i++){
            $nombre="foto".$i;
            if (null!=$inmueble[$nombre]){
                array_push($arImg,$inmueble[$nombre]);
            }
        }
    }

    ?>

<!-- Imprimir imagenes -->

<div id="myCarousel" class="carousel slide bg-inverse w-50 ml-auto mr-auto" data-ride="carousel">
<div class="carousel-inner" role="listbox">

    <div class="carousel-item active zoomMuyGrande">
        <img class="d-block w-100" src=<?php echo $arImg[0];?> >
        <div class="carousel-caption">
        </div>
    </div>

    <?php foreach($arImg as $img): ?>
    <div class="carousel-item zoomMuyGrande">

```

```

<img class="d-block w-100" src=<?php echo $img;?> >
<div class="carousel-caption">
</div>
</div>
<?php endforeach; ?>

</div>

<a class="carousel-control-prev" href="#myCarousel" role="button" data-slide="prev">
<span class="carousel-control-prev-icon" aria-hidden="true"></span>
<span class="sr-only">Anterior</span>
</a>
<a class="carousel-control-next" href="#myCarousel" role="button" data-slide="next">
<span class="carousel-control-next-icon" aria-hidden="true"></span>
<span class="sr-only">Siguiente</span>
</a>
</div>

<!-- Datos a imprimir --&gt;

&lt;table class="table table-striped"&gt;
&lt;tbody&gt;
&lt;tr class="fila"&gt;
&lt;th scope="row"&gt;Referencia:&lt;/th&gt;
&lt;td&gt;&lt;?php echo $inmueble['id'];?&gt;&lt;/td&gt;
&lt;th&gt;Tipo:&lt;/th&gt;
&lt;td&gt;&lt;?php echo $inmueble['tipo'];?&gt;&lt;/td&gt;
&lt;/tr&gt;
&lt;tr class="fila"&gt;
&lt;th scope="row"&gt;Alquiler:&lt;/th&gt;
&lt;td&gt;&lt;?php echo $inmueble['alquiler'];?&gt;&lt;/td&gt;
&lt;th&gt;Venta:&lt;/th&gt;
&lt;td&gt;&lt;?php echo $inmueble['venta'];?&gt;&lt;/td&gt;
&lt;/tr&gt;
&lt;tr class="fila"&gt;
&lt;th scope="row"&gt;Habitaciones:&lt;/th&gt;
&lt;td&gt;&lt;?php echo $inmueble['habitaciones'];?&gt;&lt;/td&gt;
&lt;th&gt;Metros:&lt;/th&gt;
&lt;td&gt;&lt;?php echo $inmueble['metros'];?&gt;&lt;/td&gt;
&lt;/tr&gt;
&lt;tr class="fila"&gt;
&lt;th scope="row"&gt;Ciudad:&lt;/th&gt;
&lt;td&gt;&lt;?php echo $inmueble['ciudad'];?&gt;&lt;/td&gt;
&lt;th&gt;Provincia:&lt;/th&gt;
&lt;td&gt;&lt;?php echo $inmueble['provincia'];?&gt;&lt;/td&gt;
&lt;/tr&gt;
&lt;tr class="fila"&gt;
&lt;th scope="row"&gt;Calle:&lt;/th&gt;
&lt;td&gt;&lt;?php echo $inmueble['calle'];?&gt;&lt;/td&gt;
&lt;th&gt;Numero:&lt;/th&gt;
&lt;td&gt;&lt;?php echo $inmueble['numero'];?&gt;&lt;/td&gt;
&lt;/tr&gt;
&lt;tr class="fila"&gt;
&lt;th scope="row"&gt;Altura:&lt;/th&gt;
&lt;td&gt;&lt;?php echo $inmueble['piso'];?&gt;&lt;/td&gt;
&lt;th&gt;Ascensor:&lt;/th&gt;
&lt;/tr&gt;
</pre>

```

```

<td><?php if('nu'!=$inmueble['ascensor']){
    echo $inmueble['ascensor'];
}else{
    echo "No";
}?></td>
</tr>

<tr class="fila">
<th scope="row">CP:</th>
<td><?php echo $inmueble['cp'];?></td>
<th scope="row">Telefono</th>
<td><?php echo $inmueble['telefonoPropietario'];?></td>

</tr>

</tbody>
</table>

<table class="table table-striped fila">
<tbody>
<th scope="row">Comentario:</th>
<td><?php echo $inmueble['descripcion'];?></td>

</tbody>
</table>
<!-- Footer -->
<footer class="py-1 bg-dark ">
<div class="container">
<div class="row">
<div class="col-3">
<p class="text-primary "> Daniel Ivaylov</p>
</div>

<div class="col-3">
<a href="index.php"><button type="button" class="btn btn-primary">Inicio</button></a>
</div>

<div class="col-3">
<p class="text-primary ">Inmobiliaria en: C/San Sebastian 5 Valencia CP 46019</p>
</div>

<div class="col-3">
<marquee class="text-primary " behavior="scroll" direction="left"> Informacion al 961 616 161 </marquee>
</div>

</div>
</div>
</footer>

<!-- Bootstrap JavaScript -->
<script src="js/jquery-3.3.1.min.js"></script>
<script src="js/bootstrap.js"></script>

</div>
</body>

```

```

</html>

registrarAdministrador.views.php
<?php
ob_start();

require __DIR__ . "/partials/inmobiliaria.part.php";

?>

<div class="container m-5">
<div class="row text-left">

<form action="registrarAdministrador.php" method="POST" enctype="multipart/form-data">
<h1>Registro de administrador.</h1>

<div class="form-group">
<label for="formGroupExampleInput">Usuario</label>
<input type="text" class="form-control" id="nombre" name="usuario" placeholder="Ejemplo: TzarDaniel"
required>
</div>

<div class="form-group">
<label for="formGroupExampleInput">Nombre</label>
<input type="text" class="form-control" id="nombre" name="nombre" placeholder="Ejemplo: Daniel" required>
</div>
<div class="form-group">
<label for="formGroupExampleInput2">Apellido</label>
<input type="text" class="form-control" id="apellido" name="apellido" placeholder="Ejemplo: Gerchev" required>
</div>
<div class="form-group">
<label for="formGroupExampleInput2">Telefono</label>
<input type="text" class="form-control" id="telefono" name="telefono" placeholder="Ejemplo:653366987"
required>
</div>

<div class="form-group">
<label for="exampleInputEmail1">Email</label>
<input type="email" class="form-control" aria-describedby="emailHelp" name="correo" placeholder="Ejemplo:
divaylov@ieslavereda.es" required>
</div>
<div class="form-group">
<label for="exampleInputPassword1">Contraseña</label>
<input type="password" class="form-control" name="contra" placeholder="Introduce contraseña" required>
</div>

<input type="submit" value="Enviar">
</form>

</div>
</div>

```

```

<!-- Footer -->
<footer class="py-1 bg-dark ">
  <div class="container">
    <div class="row">
      <div class="col-3">
        <p class="text-primary "> Daniel Ivaylov</p>
      </div>

      <div class="col-3">
        <a href="index.php"><button type="button" class="btn btn-primary">Inicio</button></a>
      </div>

      <div class="col-3">
        <p class="text-primary ">Inmobiliaria en: C/San Sebastian 5 Valencia CP 46019</p>
      </div>

      <div class="col-3">
        <marquee class="text-primary " behavior="scroll" direction="left"> Informacion al 961 616 161 </marquee>
      </div>

    </div>
  </div>
</footer>

```

```

<!-- Bootstrap JavaScript -->
<script src="js/jquery-3.3.1.min.js"></script>
<script src="js/bootstrap.js"></script>

</div>
</body>

</html>
registrarUsuario.views.php
<?php
require __DIR__ . "/partials/inmobiliaria.part.php";
ob_start();

?>
<div class="container m-5">
  <div class="row text-left">

<form action="registrarUsuario.php" method="POST" enctype="multipart/form-data">
  <h1>Registrate para anunciar tus inmuebles totalmente gratis!</h1>

  <div class="form-group">
    <label class="text-left" for="formGroupExampleInput">Usuario</label>
    <input type="text" class="form-control" id="nombre" name="usuario" placeholder="Ejemplo: TzarDaniel"
required>
  </div>

  <div class="form-group">
    <label class="text-left" for="formGroupExampleInput">Nombre</label>
    <input type="text" class="form-control" id="nombre" onblur="comprobarLetrasNombre(); name="nombre"
placeholder="Ejemplo: Daniel" required>
  </div>

```

```

</div>
<div class="form-group">
    <label class="text-left" for="formGroupExampleInput2">Apellido</label>
    <input type="text" class="form-control" id="apellido" onblur="comprobarLetrasApellido();" name="apellido"
placeholder="Ejemplo: Gerchev" required>
</div>
<div class="form-group">
    <label class="text-left" for="formGroupExampleInput2">Telefono</label>
    <input type="text" class="form-control" id="telefono" onblur="comprobarTelefono();" name="telefono"
placeholder="Ejemplo:653366987" required>
</div>

<div class="form-group">
    <label class="text-left" for="exampleInputEmail1">Email</label>
    <input type="email" class="form-control" aria-describedby="emailHelp" name="correo" placeholder="Ejemplo:
divaylov@ieslavereda.es" required>
    <small class="form-text text-muted">Nunca vamos a dar su informacion a terceros.</small>
</div>
<div class="form-group">
    <label class="text-left" for="exampleInputPassword1">Contraseña</label>
    <input type="password" class="form-control" name="contra" placeholder="Introduce contraseña" required>
</div>

<input type="submit" value="Enviar">
</form>

</div>
</div>

```

```

<SCRIPT LANGUAGE="Javascript">
let arCampos=document.getElementsByTagName('input');

let regNumero=/^\d+$/;
let regLetras=/^D+$/;

function comprobarLetrasNombre(){
    if (regLetras.test(arCampos[1].value)) {
        arCampos[1].style.background="green";
    }else{
        arCampos[1].style.background="red";
    }
}

function comprobarLetrasApellido(){
    if (regLetras.test(arCampos[2].value)) {
        arCampos[2].style.background="green";
    }else{
        arCampos[2].style.background="red";
    }
}

function comprobarTelefono(){
    if (regNumero.test(arCampos[3].value)) {
        arCampos[3].style.background="green";
    }else{
        arCampos[3].style.background="red";
    }
}

```

```

</SCRIPT>

<!-- Footer -->
<footer class="py-1 bg-dark">
  <div class="container">
    <div class="row">
      <div class="col-3">
        <p class="text-primary"> Daniel Ivaylov</p>
      </div>

      <div class="col-3">
        <a href="index.php"><button type="button" class="btn btn-primary">Inicio</button></a>
      </div>

      <div class="col-3">
        <p class="text-primary">Inmobiliaria en: C/San Sebastian 5 Valencia CP 46019</p>
      </div>

      <div class="col-3">
        <marquee class="text-primary" behavior="scroll" direction="left"> Informacion al 961 616 161 </marquee>
      </div>

    </div>
  </div>
</footer>

<!-- Bootstrap JavaScript -->
<script src="js/jquery-3.3.1.min.js"></script>
<script src="js/bootstrap.js"></script>

</div>
</body>

</html>

<?php
ob_start();

require __DIR__."/partials/inmobiliaria.part.php";
require_once 'Connection.php';
$PDO=Connection::make();

$objetos=[];

$idPropietario=$_SESSION['id'];

if ($_SERVER['REQUEST_METHOD']=='GET') {

```

```

$statement=$PDO->prepare("SELECT * FROM inmueble where idPropietario=:idProp");
$statement->setFetchMode(PDO::FETCH_CLASS|PDO::FETCH_PROPS_LATE,"inmueble");
$statement->bindParam(':idProp', $idPropietario);
$statement->execute();
$susPisos=$statement->fetchAll();

$statement=$PDO->prepare("SELECT * FROM propietario where id=:idProp");
$statement->setFetchMode(PDO::FETCH_CLASS|PDO::FETCH_PROPS_LATE,"propietario");
$statement->bindParam(':idProp', $idPropietario);
$statement->execute();
$datosPropietario=$statement->fetchAll();
//sus pisos seran los pisos del usuario que posee y puede borrar
//print_r($datosPropietario);
}

?>

```

```

<div class="container m-2">

    <h1 class="text-center ">Estos son todos tus inmuebles, <?php echo $datosPropietario[0]["nombre"]; ?>. Si quieres
borrar alguno solo tienes que seleccionarlo. Si deseas volver atras puedes hacerlo al final de la pagina.</h1>
    <div class="row text-center">
        <?php foreach($susPisos as $piso): ?>
            <div class="col-lg-3 col-md-6 mb-4">
                <div class="card h-100">
                    <img class="card-img-top" src=<?php echo $piso['foto1'];?>>
                    <div class="card-body">
                        <h4 class="card-title">Ciudad <?php echo $piso['ciudad'];?> </h4>
                        <p class="card-text">
                            Informacion sobre inmueble:
                        </p>
                        <p class="text-left card-text">
                            Id propietario: <?php echo $piso['idPropietario'];?><br>
                            Tipo de inmueble: <?php echo $piso['tipo'];?><br>
                            Habitaciones: <?php echo $piso['habitaciones'];?><br>
                            Metros cuadrados: <?php echo $piso['metros'];?><br>
                            Altura: <?php echo $piso['piso'];?><br>
                            Numero de referencia:
                        </p>
                        <form action="usuarioBorraInmueble.php" method="POST" enctype="multipart/form-data">
                            <input type="text" readonly name="id" value=<?php echo $piso['id'];?> />
                        </div>
                        <div class="card-footer">
                            <!--class="btn btn-primary submit"-->
                            <input class="btn btn-primary" type="submit" value="Borrar inmueble"></input>
                        </div>
                    </div>
                </div>
            <?php endforeach; ?>

```

```

</div>

</div>
<div class="container m-2">

    <div class="row justify-content-around">
        <a title="Borra tus inmuebles" href="funcionesUsuario.php">
            
        </div>
    </div>

```

<!-- Footer -->

```

<footer class="py-1 bg-dark ">
    <div class="container">
        <div class="row">
            <div class="col-3">
                <p class="text-primary "> Daniel Ivaylov</p>
            </div>

            <div class="col-3">
                <a href="index.php"><button type="button" class="btn btn-primary">Inicio</button></a>
            </div>

            <div class="col-3">
                <p class="text-primary ">Inmobiliaria en: C/San Sebastian 5 Valencia CP 46019</p>
            </div>

            <div class="col-3">
                <marquee class="text-primary " behavior="scroll" direction="left"> Informacion al 961 616 161 </marquee>
            </div>
        </div>
    </div>
</footer>

<!-- Bootstrap JavaScript -->
<script src="js/jquery-3.3.1.min.js"></script>
<script src="js/bootstrap.js"></script>

</div>
</body>

</html>

Administrador.php
<?php
/**
 * Created by PhpStorm.
 * User: dani
 * Date: 19/04/19
 * Time: 17:31
 */

```

```

class Administrador extends Persona
{
}

alquilar.php
<?php

session_start();
ob_start();

require "views/alquilar.views.php";

require_once 'Connection.php';
$PDO=Connection::make();


?>
comprar.php

<?php

session_start();
ob_start();

require "views/alquilar.views.php";

require_once 'Connection.php';
$PDO=Connection::make();


?>

comprarAlquilar.php
<?php

session_start();
ob_start();

require "views/comprarAlquilar.views.php";

require_once 'Connection.php';
$PDO=Connection::make();


?>
Connection.php
<?php
/**
 * Created by PhpStorm.
 * User: dani
 * Date: 7/01/19
 * Time: 16:39
 */
class Connection
{
    public static function make(){
        try{
            $opciones=[PDO::MYSQL_ATTR_INIT_COMMAND=>"SET NAMES utf8",

```

```

    PDO::ATTR_ERRMODE=>PDO::ERRMODE_EXCEPTION,
    PDO::ATTR_PERSISTENT=>true];
$cexion=new PDO(
    'mysql:localhost=127.0.0.1;dbname=inmobiliaria2;charset=utf8',
    'root',
    '',
    '',
    $opciones);

}catch (PDOException $PDOException){
    die($PDOException->getMessage());
}
return $cexion;
}
}

contacto.php
<?php
session_start();
ob_start();

require "views/contacto.views.php";

require_once 'Connection.php';
$PDO=Connection::make();

?>
Empleado.php
<?php
/**
 * Created by PhpStorm.
 * User: dani
 * Date: 19/04/19
 * Time: 17:30
 */

class Empleado extends Persona

{
}

funcionesAdministrador.php
<?php
session_start();
ob_start();

require "views/funcionesAdministrador.views.php";
require_once 'Connection.php';
$PDO=Connection::make();

if ($_SERVER['REQUEST_METHOD']=='POST') {

    $sql="INSERT INTO empleado (usuario, nombre, apellido, telefono, correo,contra) VALUES
(:usuario,:nombre,:apellido, :telefono, :correo, :contra)";
    $usuario=$_POST['usuario'];
    $apellido=$_POST['apellido'];
    $telefono=$_POST['telefono'];
    $correo=$_POST['correo'];
    $nombre=$_POST['nombre'];
    $contra=$_POST['contra'];
}

```

```

if ($usuario!=""){
    if ($nombre!=""){
        if ($apellido!=""){
            if ($telefono!=""){
                if ($correo!=""){
                    if ($contra!=""){

                        //encriptamos la contraseña
                        $contra=hash2( $contra);

                        $statement=$PDO->prepare($sql);
                        $statement->bindParam(':usuario', $usuario);
                        $statement->bindParam(':nombre', $nombre);
                        $statement->bindParam(':apellido', $apellido);
                        $statement->bindParam(':telefono', $telefono);
                        $statement->bindParam(':correo', $correo);
                        $statement->bindParam(':contra', $contra);

                        $statement->execute();
                    ?> <script>
                        alert("El usuario se ha registrado correctamente");
                    </script>
                    <?php
                }
            }
        }
    }
}
else{
    ?> <script>
        alert("Falta contraseña");
    </script>
    <?php
}
else{
    ?> <script>
        alert("Falta correo");
    </script>
    <?php
}
else{
    ?> <script>
        alert("Falta telefono");
    </script>
    <?php
}
else{
    ?> <script>
        alert("Falta apellido");
    </script>
    <?php
}
else{
    ?> <script>
        alert("Falta nombre");
    </script>
    <?php
}
else{
    ?> <script>
        alert("falta usuario");
    </script>
    <?php
}

}

function hash2($password) {
    return password_hash($password, PASSWORD_DEFAULT, ['cost'=> 15]);
}

```

```

}

?>

funcionesEmpleado.php
<?php
session_start();
ob_start();

require "views/funcionesEmpleado.views.php";
require_once 'Connection.php';
$PDO=Connection::make();

if ($_SERVER['REQUEST_METHOD']=='POST') {

    $id=$_POST['id'];

    $sql="UPDATE inmueble SET alquiler=0 where id=:id";
    $statement=$PDO->prepare($sql);
    $statement->bindParam(':id', $id);

    $statement->execute();

    ?> <script>
        alert("El piso se ha alquilado");
    </script>
    <?php

}

?>
funcionesEmpleadoVender.php
<?php
session_start();
ob_start();

require "views/funcionesEmpleado.views.php";
require_once 'Connection.php';
$PDO=Connection::make();

if ($_SERVER['REQUEST_METHOD']=='POST') {

    $id=$_POST['id'];

    $sql="DELETE from inmueble where inmueble.id=:id";
    $statement=$PDO->prepare($sql);
    $statement->bindParam(':id', $id);

    $statement->execute();

    ?> <script>
        alert("Se ha completado la peticion.");
    </script>
    <?php

}

}

```

```

?>
funcionesUsuario.php
<?php
session_start();

require "views/funcionesUsuario.views.php";

require_once 'Connection.php';
$PDO=Connection::make();

$idPropietario=$_SESSION['id'];

if ($_SERVER['REQUEST_METHOD']=='POST') {

    $sql="INSERT INTO inmueble (tipo , alquiler, venta, habitaciones, metros,provincia, ciudad, calle, cp, numero,
puerta, piso, idPropietario,telefonoPropietario,descripcion,ascensor,foto1,foto2,foto3,foto4,foto5) VALUES (:tipo ,
:alquiler, :venta, :habitaciones, :metros, :provincia, :ciudad, :calle, :cp, :numero, :puerta, :piso, :idPropietario,
:telefonoPropietario, :descripcion, :ascensor, :foto1,:foto2, :foto3, :foto4,:foto5)";
    $tipo=$_POST['tipo'];
    $habitaciones=$_POST['habitaciones'];
    $metros=$_POST['metros'];
    $provincia=$_POST['provincia'];
    $ciudad=$_POST['ciudad'];
    $calle=$_POST['calle'];
    $cp=$_POST['cp'];
    $numero=$_POST['numero'];
    $puerta=$_POST['puerta'];
    $piso=$_POST['piso'];
    $telefonoPropietario=$_POST['telefonoPropietario'];
    $descripcion=$_POST['descripcion'];
    $ascensor=$_POST['ascensor'];

    if(!isset($alquiler)){
        $alquiler=$_POST['alquiler'];
    }
    if(!isset($venta)){
        $venta=$_POST['venta'];
    }
}

//EL ID ESTA RESERVADO DE ANTES GUARDADO EN SESION COMO EL NOMBRE DEL USUARIO.

```

```

//alquiler y venta pueden no haberse introducido por lo tanto no hacemos comprobacion hasta despues y
comprobamos que tenga por lo menos la primera foto que es obligatoria.

if ($tipo!=""){
    if ($habitaciones!=""){
        if ($metros!=""){
            if ($provincia!=""){
                if ($ciudad!=""){
                    if ($calle!=""){
                        if ($cp!=""){
                            if ($numero!=""){
                                if ($puerta!=""){
                                    if ($piso!=""){
                                        if ($descripcion!=""){
                                            if ($ascensor!=""){
                                                if ($telefonoPropietario!=""){


```

```

$name1=$_FILES['foto1']['name'];
$name2=$_FILES['foto2']['name'];
$name3=$_FILES['foto3']['name'];
$name4=$_FILES['foto4']['name'];
$name5=$_FILES['foto5']['name'];
$ foto1="imagenes/".$name1;
$ foto2="imagenes/".$name2;
$ foto3="imagenes/".$name3;
$ foto4="imagenes/".$name4;
$ foto5="imagenes/".$name5;
$idUnico = time();

if (is_file($foto1) === true) {
    $name1 = $idUnico . '_f1_' . $name1;
}if (move_uploaded_file($_FILES['foto1']['tmp_name'], $foto1)) {

    $statement = $PDO->prepare($sql);
    $statement->bindParam(':tipo', $tipo);
    $statement->bindParam(':habitaciones', $habitaciones);
    $statement->bindParam(':metros', $metros);
    $statement->bindParam(':provincia', $provincia);
    $statement->bindParam(':ciudad', $ciudad);
    $statement->bindParam(':calle', $calle);
    $statement->bindParam(':telefonoPropietario', $telefonoPropietario);
    $statement->bindParam(':cp', $cp);
    $statement->bindParam(':numero', $numero);
    $statement->bindParam(':puerta', $puerta);
    $statement->bindParam(':piso', $piso);
    $statement->bindParam(':descripcion', $descripcion);
    $statement->bindParam(':ascensor', $ascensor);
    $statement->bindParam(':foto1', $foto1);

    $statement->bindParam(':idPropietario', $idPropietario);

    if (isset($venta)) {
        $statement->bindParam(':venta', $venta);
    }

    if (isset($alquiler)) {
        $statement->bindParam(':alquiler', $alquiler);
    }
}

if (is_file($foto2) === true) {
    $name2 = $idUnico . '_f2_' . $name2;
}if (move_uploaded_file($_FILES['foto2']['tmp_name'], $foto2)) {
    $statement->bindParam(':foto2', $foto2);
}
}else{
    $foto2=null;
    $statement->bindParam(':foto2', $foto2);
}

if (is_file($foto3) === true) {
    $name3 = $idUnico . '_f3_' . $name3;
}if (move_uploaded_file($_FILES['foto3']['tmp_name'], $foto3)) {
    $statement->bindParam(':foto3', $foto3);
}

```

```

        }
    }else{
        $foto3=null;
        $statement->bindParam(':foto3', $foto3);

    }

    if (is_file($foto4) === true) {
        $name4 = $idUnico . '_f4_' . $name4;
        if (move_uploaded_file($_FILES['foto4']['tmp_name'], $foto4)) {
            $statement->bindParam(':foto4', $foto4);
        }
    }else{
        $foto4=null;
        $statement->bindParam(':foto4', $foto4);

    }

    if (is_file($foto5) === true) {
        $name5 = $idUnico . '_f5_' . $name5;
        if (move_uploaded_file($_FILES['foto5']['tmp_name'], $foto5)) {
            $statement->bindParam(':foto5', $foto5);
        }
    }else{
        $foto5=null;
        $statement->bindParam(':foto5', $foto5);

    }

}

// ver la insercion echo $statement->queryString;
$statement->execute();
?>
<script>
    alert("El piso se ha registrado");
</script>
<?php
}else{
?>
<script>
    alert("Falta foto ");
</script>
<?php
?

}else{
?>
<script>
    alert("Falta telefono ");
</script>
<?php
?

}else{
?>
<script>
    alert("Falta ascensor");
</script>
<?php
?
}

```

```
        }

    }else{
    ?>
<script>
    alert("Falta descripcion");
</script>
<?php
}

}else{
    ?>
<script>
    alert("Falta piso");
</script>
<?php
}

}else{
    ?>
<script>
    alert("Falta puerta");
</script>
<?php
}

}else{
    ?>
<script>
    alert("Falta numero ");
</script>
<?php
}

}else{
    ?>
<script>
    alert("Falta cp");
</script>
<?php
}

}else{
    ?>
<script>
    alert("Falta calle");
</script>
<?php
}

}else{
    ?>
<script>
    alert("Falta ciudad");
</script>
<?php
}
```

```

        }else{
    ?>
    <script>
        alert("Falta provincia");
    </script>
    <?php
    }
}else{
    ?>
    <script>
        alert("Falta metros");
    </script>
    <?php
}
}

}else{
    ?>
    <script>
        alert("Falta numero habitaciones");
    </script>
    <?php
}
}

}else{
    ?>
    <script>
        alert("Falta tipo");
    </script>
    <?php
}
}

?>

imprimirAlquiler.php
<?php

session_start();
ob_start();

require "views/imprimirAlquiler.views.php";

require_once 'Connection.php';
$PDO=Connection::make();

?>
imprimirComprar.php
<?php

session_start();
ob_start();

require "views/imprimirComprar.views.php";

require_once 'Connection.php';
$PDO=Connection::make();

?>

```

```

index.php
<?php

session_start();
ob_start();

require "views/index.views.php";

require_once 'Connection.php';
$PDO=Connection::make();

?>
inmueble.php
<?php
/**
 * Created by PhpStorm.
 * User: dani
 * Date: 2/05/19
 * Time: 12:40
 */
class inmueble
{
    protected $tipo;
    protected $id;
    protected $alquiler;
    protected $venta;
    protected $habitaciones;
    protected $metros;
    protected $provincia;
    protected $ciudad;
    protected $calle;
    protected $cp;
    protected $numero;
    protected $puerta;
    protected $piso;
    protected $telefonoPropietario;
    protected $descripcion;
    protected $ascensor;
    protected $foto1;
    protected $foto2;
    protected $foto3;
    protected $foto4;
    protected $foto5;

    /**
     * inmueble constructor.
     * @param $tipo
     * @param $id
     * @param $alquiler
     * @param $venta
     * @param $habitaciones
     * @param $metros
     * @param $provincia
     * @param $ciudad
     * @param $calle
     * @param $cp
}

```

```

 * @param $numero
 * @param $puerta
 * @param $piso
 * @param $telefonoPropietario
 * @param $descripcion
 * @param $ascensor
 * @param $foto1
 * @param $foto2
 * @param $foto3
 * @param $foto4
 * @param $foto5
 */
public function __construct($tipo=null, $id=null, $alquiler=null, $venta=null, $habitaciones=null, $metros=null,
$provincia=null, $ciudad=null, $calle=null, $cp=null, $numero=null, $puerta=null, $piso=null,
$telefonoPropietario=null, $descripcion=null, $ascensor=null, $foto1=null, $foto2=null, $foto3=null, $foto4=null,
$foto5=null)
{
    $this->tipo = $tipo;
    $this->id = $id;
    $this->alquiler = $alquiler;
    $this->venta = $venta;
    $this->habitaciones = $habitaciones;
    $this->metros = $metros;
    $this->provincia = $provincia;
    $this->ciudad = $ciudad;
    $this->calle = $calle;
    $this->cp = $cp;
    $this->numero = $numero;
    $this->puerta = $puerta;
    $this->piso = $piso;
    $this->telefonoPropietario = $telefonoPropietario;
    $this->descripcion = $descripcion;
    $this->ascensor = $ascensor;
    $this->foto1 = $foto1;
    $this->foto2 = $foto2;
    $this->foto3 = $foto3;
    $this->foto4 = $foto4;
    $this->foto5 = $foto5;
}

/**
 * @return null
 */
public function getTipo()
{
    return $this->tipo;
}

/**
 * @param null $tipo
 */
public function setTipo($tipo): void
{
    $this->tipo = $tipo;
}

/**
 * @return null
 */

```

```

public function getId()
{
    return $this->id;
}

/*
 * @param null $id
 */
public function setId($id): void
{
    $this->id = $id;
}

/*
 * @return null
 */
public function getAlquiler()
{
    return $this->alquiler;
}

/*
 * @param null $alquiler
 */
public function setAlquiler($alquiler): void
{
    $this->alquiler = $alquiler;
}

/*
 * @return null
 */
public function getVenta()
{
    return $this->venta;
}

/*
 * @param null $venta
 */
public function setVenta($venta): void
{
    $this->venta = $venta;
}

/*
 * @return null
 */
public function getHabitaciones()
{
    return $this->habitaciones;
}

/*
 * @param null $habitaciones
 */
public function setHabitaciones($habitaciones): void
{
    $this->habitaciones = $habitaciones;
}

```

```

}

/**
 * @return null
 */
public function getMetros()
{
    return $this->metros;
}

/**
 * @param null $metros
 */
public function setMetros($metros): void
{
    $this->metros = $metros;
}

/**
 * @return null
 */
public function getProvincia()
{
    return $this->provincia;
}

/**
 * @param null $provincia
 */
public function setProvincia($provincia): void
{
    $this->provincia = $provincia;
}

/**
 * @return null
 */
public function getCiudad()
{
    return $this->ciudad;
}

/**
 * @param null $ciudad
 */
public function setCiudad($ciudad): void
{
    $this->ciudad = $ciudad;
}

/**
 * @return null
 */
public function getCalle()
{
    return $this->calle;
}

```

```

    * @param null $calle
    */
public function setCalle($calle): void
{
    $this->calle = $calle;
}

/**
 * @return null
 */
public function getCp()
{
    return $this->cp;
}

/**
 * @param null $cp
 */
public function setCp($cp): void
{
    $this->cp = $cp;
}

/**
 * @return null
 */
public function getNumero()
{
    return $this->numero;
}

/**
 * @param null $numero
 */
public function setNumero($numero): void
{
    $this->numero = $numero;
}

/**
 * @return null
 */
public function getPuerta()
{
    return $this->puerta;
}

/**
 * @param null $puerta
 */
public function setPuerta($puerta): void
{
    $this->puerta = $puerta;
}

/**
 * @return null
 */
public function getPiso()

```

```

{
    return $this->piso;
}

/**
 * @param null $piso
 */
public function setPiso($piso): void
{
    $this->piso = $piso;
}

/**
 * @return null
 */
public function getTelefonoPropietario()
{
    return $this->telefonoPropietario;
}

/**
 * @param null $telefonoPropietario
 */
public function setTelefonoPropietario($telefonoPropietario): void
{
    $this->telefonoPropietario = $telefonoPropietario;
}

/**
 * @return null
 */
public function getDescripcion()
{
    return $this->descripcion;
}

/**
 * @param null $descripcion
 */
public function setDescripcion($descripcion): void
{
    $this->descripcion = $descripcion;
}

/**
 * @return null
 */
public function getAscensor()
{
    return $this->ascensor;
}

/**
 * @param null $ascensor
 */
public function setAscensor($ascensor): void
{
    $this->ascensor = $ascensor;
}

```

```

/*
 * @return null
 */
public function getFoto1()
{
    return $this->foto1;
}

/**
 * @param null $foto1
 */
public function setFoto1($foto1): void
{
    $this->foto1 = $foto1;
}

/*
 * @return null
 */
public function getFoto2()
{
    return $this->foto2;
}

/**
 * @param null $foto2
 */
public function setFoto2($foto2): void
{
    $this->foto2 = $foto2;
}

/*
 * @return null
 */
public function getFoto3()
{
    return $this->foto3;
}

/**
 * @param null $foto3
 */
public function setFoto3($foto3): void
{
    $this->foto3 = $foto3;
}

/*
 * @return null
 */
public function getFoto4()
{
    return $this->foto4;
}

/**
 * @param null $foto4
*/

```

```

    */
public function setFoto4($foto4): void
{
    $this->foto4 = $foto4;
}

/**
 * @return null
 */
public function getFoto5()
{
    return $this->foto5;
}

/**
 * @param null $foto5
 */
public function setFoto5($foto5): void
{
    $this->foto5 = $foto5;
}

}

loguearAdministrador.php

<?php

session_start();
ob_start();

require "views/loguearAdministrador.views.php";

require_once 'Connection.php';
$PDO=Connection::make();

if ($_SERVER['REQUEST_METHOD']=='POST') {

    $sql = "SELECT contra,id from administrador where usuario=:usuario";

    $usuario=$_POST['usuario'];
    $contra=$_POST['contra'];

    $statement=$PDO->prepare($sql);
    $statement->bindParam(':usuario', $usuario);
    $statement->execute();
    $persona = $statement->fetch(PDO::FETCH_ASSOC);

    if (!$persona){
        //no hay usuario que se llame asi
        ?><script>
            alert("No se ha encontrado usuario con los datos indicados");
        </script>
        <?php
    }

} else if (password_verify($contra, $persona["contra"])){
    $_SESSION["idAdministrador"]=$persona['id'];
}

```

```

        header('Location:/inmobiliaria/funcionesAdministrador.php');
    }else if(!password_verify($contra, $persona['contra'])){
        ?> <script>
            alert("La contraseña no coincide, vuelva a intentarlo");
        </script>
        <?php
    }

}

?>
loguearEmpleado.php

<?php

session_start();
ob_start();

require "views/loguearAdministrador.views.php";

require_once 'Connection.php';
$PDO=Connection::make();

if ($_SERVER['REQUEST_METHOD']=='POST') {

    $sql = "SELECT contra,id from administrador where usuario=:usuario";

    $usuario=$_POST['usuario'];
    $contra=$_POST['contra'];

    $statement=$PDO->prepare($sql);
    $statement->bindParam(":usuario", $usuario);
    $statement->execute();
    $persona = $statement->fetch(PDO::FETCH_ASSOC);

    if (!$persona){
        //no hay usuario que se llame asi
        ?><script>
            alert("No se ha encontrado usuario con los datos indicados");
        </script>
        <?php
    }

} else if (password_verify($contra, $persona["contra"])){
    $_SESSION["idAdministrador"]=$persona['id'];

    header('Location:/inmobiliaria/funcionesAdministrador.php');
} else if(!password_verify($contra, $persona['contra'])){
    ?> <script>
        alert("La contraseña no coincide, vuelva a intentarlo");
    </script>
    <?php
}
}

```

```

?>
loguearUsuario.php
<?php

session_start();
ob_start();
require "views/loguearUsuario.views.php";

require_once 'Connection.php';
$PDO=Connection::make();

if ($_SERVER['REQUEST_METHOD']=='POST') {

    $sql = "SELECT contra,id from propietario where usuario=:usuario";

    $usuario=$_POST['usuario'];
    $contra=$_POST['contra'];

    $statement=$PDO->prepare($sql);
    $statement->bindParam(':usuario', $usuario);
    $statement->execute();
    $persona = $statement->fetch(PDO::FETCH_ASSOC);

    if (!$persona){
        //no hay usuario que se llame asi
        ?><script>
            alert("No se ha encontrado usuario con los datos indicados");
        </script>
        <?php
    }

    }else if (password_verify($contra, $persona["contra"])){
        $_SESSION["id"]=$persona['id'];

        //las contraseñas coinciden y redirigimos a funciones de usuario
        header('Location:funcionesUsuario.php');
        exit();
    }else if(!password_verify($contra, $persona['contra'])){
        ?> <script>
            alert("La contraseña no coincide, vuelva a intentarlo");
        </script>
        <?php
    }
}

?>
masInformacion.php
<?php

session_start();

require "views/masInformacion.views.php";

require_once 'Connection.php';
$PDO=Connection::make();

?>

```

```

masInformacionEmpleado.php
<?php

session_start();

require "views/masInformacionEmpleado.views.php";

require_once 'Connection.php';
$PDO=Connection::make();

?>
Persona.php
<?php
/**
 * Created by PhpStorm.
 * User: dani
 * Date: 19/04/19
 * Time: 17:25
 */

class Persona
{

    protected $id;
    protected $usuario;
    protected $nombre;
    protected $telefono;
    protected $correo;
    protected $contra;
    protected $apellido;

    /**
     * Persona constructor.
     * @param $id
     * @param $usuario
     * @param $nombre
     * @param $telefono
     * @param $correo
     * @param $contra
     * @param $dni
     */
    public function __construct($id=null, $usuario=null, $nombre=null, $telefono=null, $correo=null, $contra=null,
$apellido=null)
    {
        $this->id = $id;
        $this->usuario = $usuario;
        $this->nombre = $nombre;
        $this->telefono = $telefono;
        $this->correo = $correo;
        $this->contra = $contra;
        $this->apellido = $apellido;
    }

    /**
     * @return mixed
     */
    public function getId()
    {
        return $this->id;
    }
}

```

```

}

/**
 * @param mixed $id
 */
public function setId($id): void
{
    $this->id = $id;
}

/**
 * @return mixed
 */
public function getUsuario()
{
    return $this->usuario;
}

/**
 * @param mixed $usuario
 */
public function setUsuario($usuario): void
{
    $this->usuario = $usuario;
}

/**
 * @return mixed
 */
public function getNombre()
{
    return $this->nombree;
}

/**
 * @param mixed $nombre
 */
public function setNombre($nombre): void
{
    $this->nombree = $nombre;
}

/**
 * @return mixed
 */
public function getTelefono()
{
    return $this->telefono;
}

/**
 * @param mixed $telefono
 */
public function setTelefono($telefono): void
{
    $this->telefono = $telefono;
}

```

```

    * @return mixed
   */
public function getCorreo()
{
    return $this->correo;
}

< /**
 * @param mixed $correo
 */
public function setCorreo($correo): void
{
    $this->correo = $correo;
}

< /**
 * @return mixed
 */
public function getContra()
{
    return $this->contra;
}

< /**
 * @param mixed $contra
 */
public function setContra($contra): void
{
    $this->contra = $contra;
}

< /**
 * @return null
 */
public function getApellido()
{
    return $this->apellido;
}

< /**
 * @param null $apellido
 */
public function setApellido($apellido): void
{
    $this->apellido = $apellido;
}

}

}

```

### Propietario.php

```

<?php
/**
* Created by PhpStorm.
* User: dani
* Date: 19/04/19
* Time: 17:29

```

```

*/
class Propietario extends Persona
{
}

registrarAdministrador.php
<?php
session_start();

require "views/registrarAdministrador.views.php";
require_once 'Connection.php';
$PDO=Connection::make();

if ($_SERVER['REQUEST_METHOD']=='POST') {

    $sql="INSERT INTO administrador (usuario, nombre, apellido, telefono, correo,contra) VALUES
(:usuario,:nombre,:apellido, :telefono, :correo,:contra)";
    $usuario=$_POST['usuario'];
    $apellido=$_POST['apellido'];
    $telefono=$_POST['telefono'];
    $correo=$_POST['correo'];
    $nombre=$_POST['nombre'];
    $contra=$_POST['contra'];

    if ($usuario!=""){
        if ($nombre!=""){
            if ($apellido!=""){
                if ($telefono!=""){
                    if ($correo!=""){
                        if ($contra!=""){

                            //encriptamos la contraseña
                            $contra=hash2( $contra);

                            $statement=$PDO->prepare($sql);
                            $statement->bindParam(':usuario', $usuario);
                            $statement->bindParam(':nombre', $nombre);
                            $statement->bindParam(':apellido', $apellido);
                            $statement->bindParam(':telefono', $telefono);
                            $statement->bindParam(':correo', $correo);
                            $statement->bindParam(':contra', $contra);

                            $statement->execute();
                            ?> <script>
                                alert("El usuario se ha registrado correctamente");
                            </script>
                            <?php
                                header('Location: loguearUsuario.php');

                            }else{
                                ?> <script>
                                    alert("Falta contraseña");
                                </script>
                                <?php
                            }else{
                                ?> <script>
                                    alert("Falta correo");
                                </script>
                            }
                        }
                    }
                }
            }
        }
    }
}

```

```

        </script>
    <?php      }
}else{
    ?> <script>
        alert("Falta telefono");
    </script>
<?php      }
}else{
    ?> <script>
        alert("Falta apellido");
    </script>
<?php      }
}else{
    ?> <script>
        alert("Falta nombre");
    </script>
<?php      }
}else{
    ?> <script>
        alert("Falta usuario");
    </script>
<?php  }
}

function hash2($password) {
    return password_hash($password, PASSWORD_DEFAULT, ['cost'=> 15]);
}

?>
registrarUsuario.php
<?php
session_start();

require "views/registrarUsuario.views.php";
require_once 'Connection.php';
$PDO=Connection::make();

if ($_SERVER['REQUEST_METHOD']=='POST') {

    $sql="INSERT INTO propietario (usuario, nombre, apellido, telefono, correo,contra) VALUES
(:usuario,:nombre,:apellido, :telefono, :correo, :contra)";
    $usuario=$_POST['usuario'];
    $apellido=$_POST['apellido'];
    $telefono=$_POST['telefono'];
    $correo=$_POST['correo'];
    $nombre=$_POST['nombre'];
    $contra=$_POST['contra'];

    if ($usuario!=""){
        if ($nombre!=""){
            if ($apellido!=""){
                if ($telefono!=""){
                    if ($correo!=""){
                        if ($contra!=""){

                            //encriptamos la contraseña
                            $contra=hash2( $contra);

```

```

$statement=$PDO->prepare($sql);
$statement->bindParam(':usuario', $usuario);
$statement->bindParam(':nombre', $nombre);
$statement->bindParam(':apellido', $apellido);
$statement->bindParam(':telefono', $telefono);
$statement->bindParam(':correo', $correo);
$statement->bindParam(':contra', $contra);

$statement->execute();
?> <script>
    alert("El usuario se ha registrado correctamente");
</script>
<?php
header('Location: loguearUsuario.php');

}else{
?> <script>
    alert("Falta contraseña");
</script>
<?php
}else{
?> <script>
    alert("Falta correo");
</script>
<?php
}else{
?> <script>
    alert("Falta telefono");
</script>
<?php
}else{
?> <script>
    alert("Falta apellido");
</script>
<?php
}else{
?> <script>
    alert("Falta nombre");
</script>
<?php
}else{
?> <script>
    alert("Falta usuario");
</script>
<?php
}
}

function hash2($password) {
    return password_hash($password, PASSWORD_DEFAULT, ['cost'=> 15]);
}

?>
usuarioBorraInmueble.php
<?php

```

```

session_start();
ob_start();

require "views/usuarioBorraInmueble.views.php";

require_once 'Connection.php';
$PDO=Connection::make();

if ($_SERVER['REQUEST_METHOD']=='POST') {

    $id=$_POST['id'];

    $sql="DELETE from inmueble where inmueble.id=:id";
    $statement=$PDO->prepare($sql);
    $statement->bindParam(':id', $id);

    $statement->execute();

    ?> <script>
        alert("Se ha completado la petición.");
    </script>
    <?php
        header('Location: usuarioBorraInmueble.php');

    }

?>
mi.css

* {
    margin: 0; padding: 0;
}

.zoom:hover {
    transform:scale(1.1, 1.1);
}

.zoomGrande:hover {
    transform:scale(1.3, 1.3);
}

.zoomMuyGrande:hover {
    transform:scale(1.5, 1.5);
}

.helicoptero:hover {
    transform: rotate(360deg);
    transition: 3s;
}

```

```

h1 {
    font-family: 'Pacifico', cursive;
}

.circulito:hover{
    border-radius: 50%;
}

.fila {
    animation-duration: 3s;
}

.fila:hover {
    background-color: dodgerblue;
    color: black;
    -webkit-box-shadow: 0px 1px 37px 9px rgba(0,0,0,0.75);
    -moz-box-shadow: 0px 1px 37px 9px rgba(0,0,0,0.75);
    box-shadow: 0px 1px 37px 9px rgba(0,0,0,0.75);
}

.contraste:hover{
    filter:contrast(200%);
}

.arcoiris:hover{
    animation-name: colores;
    animation-duration: 10s;
}

@keyframes colores {
    0% {background-color: yellow;}
    25% {background-color: greenyellow;}
    50% {background-color: deepskyblue;}
    100% {background-color: red;}
}

.derecha{
    position:absolute;
    right:10px;
}

-- phpMyAdmin SQL Dump
-- version 4.8.3
-- https://www.phpmyadmin.net/
--
-- Servidor: localhost
-- Tiempo de generación: 27-05-2019 a las 23:27:14
-- Versión del servidor: 10.1.35-MariaDB
-- Versión de PHP: 7.2.9

```

```

SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
SET AUTOCOMMIT = 0;
START TRANSACTION;
SET time_zone = "+00:00";

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8mb4 */;

-- 
-- Base de datos: `inmobiliaria2`
-- 

-- -----
-- 

-- 
-- Estructura de tabla para la tabla `administrador`
-- 

CREATE TABLE `administrador` (
  `id` int(255) NOT NULL,
  `usuario` varchar(255) COLLATE utf8_spanish_ci NOT NULL,
  `nombre` varchar(255) COLLATE utf8_spanish_ci NOT NULL,
  `telefono` int(255) NOT NULL,
  `correo` varchar(255) COLLATE utf8_spanish_ci NOT NULL,
  `contra` varchar(255) COLLATE utf8_spanish_ci NOT NULL,
  `apellido` varchar(255) COLLATE utf8_spanish_ci NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_spanish_ci;

-- 
-- Volcado de datos para la tabla `administrador`
-- 

-- -----
-- 

INSERT INTO `administrador` (`id`, `usuario`, `nombre`, `telefono`, `correo`, `contra`, `apellido`) VALUES
(2, 'dani', 'Daniel', 666666666, 'divaylovg@ieslavereda.es',
'$2y$15$fN5FUbbj1vJch5uGYornXOxkvI04fCoXYZKJvqb/Pq4BE0osmL16q', 'Gerchev');

-- -----
-- 

-- Estructura de tabla para la tabla `empleado`
-- 

CREATE TABLE `empleado` (
  `id` int(255) NOT NULL,
  `usuario` varchar(255) COLLATE utf8_spanish_ci NOT NULL,
  `nombre` varchar(255) COLLATE utf8_spanish_ci NOT NULL,
  `telefono` int(255) NOT NULL,
  `correo` varchar(255) COLLATE utf8_spanish_ci NOT NULL,
  `contra` varchar(255) COLLATE utf8_spanish_ci NOT NULL,
  `apellido` varchar(255) COLLATE utf8_spanish_ci NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_spanish_ci;

-- 
-- Volcado de datos para la tabla `empleado`
-- 

-- 

```

```

INSERT INTO `empleado` (`id`, `usuario`, `nombre`, `telefono`, `correo`, `contra`, `apellido`) VALUES
(1, 'empleado1', 'asdf', 666666666, 'empleado1@asdf.com',
'$2y$15$xNLFDU6QQml6Y6AD4IkNuTW8CtPLG45To2BHjxe4RUjGGEjc9fOK', 'asdf'),
(2, 'empleado2', 'qqqq', 666666666, 'emp@go.com',
'$2y$15$EvqZz1321rJelh28GAuXIO0ol8v5FQHKyj7wC6tPqvKzjAuDcv53e', 'dani'),
(3, 'empleado3', 'asdfasdf', 666666, 'empleado3@gmail.com',
'$2y$15$Ni1HLvtN1D5nkDhlVFvFJ.CdPH8Zb5NrI8tEAff9Vhe0k2s2oiM2', 'asdf'),
(4, 'empleado4', 'qqqq', 652222222, 'empleado4@ies.com',
'$2y$15$flff2n5nVGfmFRHYF5rFB.YnchObUYtutmfM8DakbexWaLUrekQ7G', 'asdf');

```

---

```

-- 
-- Estructura de tabla para la tabla `inmueble`
-- 
```

```

CREATE TABLE `inmueble` (
`id` int(4) NOT NULL,
`tipo` varchar(255) COLLATE utf8_spanish_ci NOT NULL,
`alquiler` int(255) DEFAULT NULL,
`venta` int(255) DEFAULT NULL,
`habitaciones` int(10) NOT NULL,
`metros` int(255) NOT NULL,
`provincia` varchar(255) COLLATE utf8_spanish_ci NOT NULL,
`ciudad` varchar(255) COLLATE utf8_spanish_ci NOT NULL,
`calle` varchar(255) COLLATE utf8_spanish_ci NOT NULL,
`cp` int(10) NOT NULL,
`numero` int(5) NOT NULL,
`puerta` int(5) NOT NULL,
`piso` int(5) NOT NULL,
`idPropietario` int(255) NOT NULL,
`telefonoPropietario` int(10) NOT NULL,
`descripcion` varchar(255) COLLATE utf8_spanish_ci NOT NULL,
`ascensor` varchar(22) COLLATE utf8_spanish_ci NOT NULL,
`foto1` varchar(50) COLLATE utf8_spanish_ci NOT NULL,
`foto2` varchar(50) COLLATE utf8_spanish_ci DEFAULT NULL,
`foto3` varchar(50) COLLATE utf8_spanish_ci DEFAULT NULL,
`foto4` varchar(50) COLLATE utf8_spanish_ci DEFAULT NULL,
`foto5` varchar(50) COLLATE utf8_spanish_ci DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_spanish_ci;

```

```

-- 
-- Volcado de datos para la tabla `inmueble`
-- 
```

```

INSERT INTO `inmueble` (`id`, `tipo`, `alquiler`, `venta`, `habitaciones`, `metros`, `provincia`, `ciudad`, `calle`, `cp`,
`numero`, `puerta`, `piso`, `idPropietario`, `telefonoPropietario`, `descripcion`, `ascensor`, `foto1`, `foto2`, `foto3`,
`foto4`, `foto5`) VALUES
(4, 'piso', 450, 0, 4, 120, 'valencia', 'Valencia', 'asdf', 46020, 45, 45, 1, 1, 666666, 'Alquilo piso en valencia', 'si',
'imagenes/5.jpg', NULL, NULL, NULL, NULL),
(5, 'chalet', 555, 555555, 1, 120, 'alava', 'Valencia', 'asdf', 3333, 44, 12, 1, 1, 666666, 'lorem', 'nu', 'imagenes/3.jpg',
'imagenes/1557074996_f2_5.jpg', NULL, NULL, NULL),
(8, 'piso', 800, 0, 4, 140, 'valencia', 'Valencia', 'asdf', 28400, 33, 1, 1, 1, 666666, 'Alquilo piso en valencia de 4
habitaciones precio negociable.', 'si', 'imagenes/1557075547_f1_5.jpg', NULL, 'imagenes/1557075547_f3_3.jpg', NULL,
'imagenes/1557075547_f5_1.jpg'),
(9, 'chalet', 555, 0, 1, 33, 'alava', 'Valencia', 'asdf', 46019, 33, 3, 1, 1, 666666, 'bonito', 'nu',
'imagenes/1557650428_f1_1.jpg', NULL, 'imagenes/1557650428_f3_3.jpg', 'imagenes/1557650428_f4_4.jpg', NULL),

```

```

(10, 'chalet', 333, 333, 1, 33, 'girona', 'Valencia', 'asdf', 3333, 33, 3, 1, 1, 3444444, 'asdf', 'nu',
'imagenes/1557650991_f1_1.jpg', 'imagenes/1557650991_f2_1.jpg', 'imagenes/1557650991_f3_1.jpg',
'imagenes/1557650991_f4_1.jpg', 'imagenes/1557650991_f5_1.jpg'),
(13, 'chalet', 333, 333, 1, 33, 'alava', 'aasdf', 'asdf', 3333, 33, 3, 1, 1, 3444444, 'asdf', 'nu', 'imagenes/1.jpg',
'imagenes/1.jpg', 'imagenes/1.jpg', 'imagenes/1.jpg', 'imagenes/1.jpg'),
(15, 'chalet', 555, 777777, 4, 133, 'madrid', 'Madrid', 'san juan', 46019, 3, 3, 1, 1, 66666666, 'Bonito piso a las afueras ',
'nu', 'imagenes/4.jpg', 'imagenes/5.jpg', 'imagenes/6.jpg', 'imagenes/3.jpg', 'imagenes/1.jpg'),
(16, 'chalet', 222, 50000, 3, 100, 'barcelona', 'Barcelona', 'san juan', 46019, 66, 3, 1, 1, 66666666, 'Bonito chalet a las
afueras de Barcelona', 'nu', 'imagenes/3.jpg', 'imagenes/4.jpg', 'imagenes/5.jpg', 'imagenes/1.jpg', 'imagenes/2.jpg'),
(17, 'chalet', 555, 777777, 1, 120, 'alava', 'Toledo', 'asdf', 3333, 33, 3, 1, 1, 666666, 'asdf', 'nu', 'imagenes/5.jpg',
'imagenes/2.jpg', 'imagenes/3.jpg', 'imagenes/2.jpg', 'imagenes/4.jpg'),
(18, 'chalet', 333, 333, 1, 33, 'alava', 'Toledo', 'To', 3333, 44, 12, 1, 1, 3444444, 'asdf', 'nu', 'imagenes/5.jpg',
'imagenes/4.jpg', 'imagenes/3.jpg', 'imagenes/2.jpg', 'imagenes/1.jpg'),
(19, 'chalet', 555, 666666, 1, 120, 'alava', 'Toledo', 'san juan', 28400, 3, 3, 1, 1, 666666, 'El mejor piso de la zona', 'null',
'imagenes/2.jpg', 'imagenes/1.jpg', 'imagenes/3.jpg', 'imagenes/4.jpg', 'imagenes/5.jpg');

```

---

```
--
```

```
-- Estructura de tabla para la tabla `propietario`
```

```
--
```

```

CREATE TABLE `propietario` (
`id` int(255) NOT NULL,
`usuario` varchar(255) COLLATE utf8_spanish_ci NOT NULL,
`nombre` varchar(255) COLLATE utf8_spanish_ci NOT NULL,
`telefono` int(255) NOT NULL,
`correo` varchar(255) COLLATE utf8_spanish_ci NOT NULL,
`contra` varchar(255) COLLATE utf8_spanish_ci NOT NULL,
`apellido` varchar(255) COLLATE utf8_spanish_ci NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_spanish_ci;

```

```
--
```

```
-- Volcado de datos para la tabla `propietario`
```

```
--
```

```

INSERT INTO `propietario` (`id`, `usuario`, `nombre`, `telefono`, `correo`, `contra`, `apellido`) VALUES
(1, 'Tzar', 'dani', 66666666, 'paco@google.com', '$2y$15$rQgmpkX4V7l/rTn804so4.B7CVf5qx61TxFFCsUADPPxNaVoz.Ai',
'dani'),
(4, 'paco', 'dani', 66666666, 'da@asdfasd.com',
'$2y$15$7oboRmqpWX9.1Sgh./CIs.KoCzkFtf7UcsTfbqbbpt9byKTKF7Au', 'dani'),
(5, 'pepe', 'qqqq', 66666666, 'pepe@gmail.com',
'$2y$15$wkhiV6MqJ7OrAAuNRT4/TuOKWCy2gvdzURo0KzrEMzvK03pUlsT6e', 'dani'),
(7, 'pedro', 'asdf', 66666666, 'pedro@gmail.com',
'$2y$15$.9D2FttPKHoE7/J29EVrOcvmRa5kDvjPo6Nmj7bzd2yayzjpai9O', 'asdf'),
(8, 'zzz', 'qqqq', 66666666, 'zzz@gmai.com',
'$2y$15$o0CI.BpzcD59LMqDQEdfO.4LeyipXp2x8faMZ59Ym.GzxgmQiKh96', 'asdf'),
(9, 'aaaaaaaa', 'aaaaaaaa', 2147483647, 'aaaaaa@hotmail.com',
'$2y$15$/C6bFsv2CF7q2nshu1hOPO1ie73vkxb9EJQZWFL78HurpgDUcKqYa', 'aaaaaaaa'),
(11, 'asdf', 'asdf', 66666666, 'asdf@google.com',
'$2y$15$HktCHj38F4JDYmbBzdezSeLvATRyczwuKtzRK9JmQHfgeJtPdMD2u', 'asdf'),
(12, 'zzzz', 'asdf', 66666666, 'zzzz@hotmail.com',
'$2y$15$S1o2n8YvNOR74dNoJ.vGcebulcz0RbtmdEd0G5qLSQ1X9pzns3dM.', 'asdf'),
(13, 'cccc', 'qqqq', 66666666, 'cccc@hotmail.com',
'$2y$15$H.pxFm5JxssDkb6B9iq3ueeklHqzOLr/oE6Aj9ebAArWOizGSiezO', 'dani'),
(14, 'tttttt', 'qqqq', 66666666, 'ttttttttt@hotmail.com',
'$2y$15$Ag9DxFCE1PrxLOsBOXmoe4yFvFTB7zmVW4YqlIU10Lydc4CYBnWW', 'dani'),
(15, 'usr', 'asdf', 66666, 'usr@hotmail.com', '$2y$15$ikXNC/d/fixCUKdEvO2aYeZsAGLsmHhtNSFuzA8ojtSko7dk9b5pm',
'asdf'),

```

```

(17, 'usr2', 'qqqq', 666666666, 'usr@go.com',
'$2y$15$vMHo0g21SMTirdX7s//4heD88yXKxrZjcd8T4cOth0HYSqp957uwC', 'asdf'),
(18, 'zzzzzdasdf', 'qqqq', 666666666, 'asdfaxxxxx@asdf.com',
'$2y$15$3QRSr0UVxh51jc6zNlu7aOicqdi5f4SSXE8uwpepl90OisQ5m/qOC', 'dani'),
(19, 'juancho', 'juan', 666666666, 'juancho@hotmail.com',
'$2y$15$W3elnf8420EV1HRn1Z8kOHGODz5pWF.FUILOIA0bMDh88Voi6USu', 'asdf');

-- 
-- Índices para tablas volcadas
-- 

-- 
-- Indices de la tabla `administrador`
-- 

ALTER TABLE `administrador`
ADD PRIMARY KEY (`id`),
ADD UNIQUE KEY `usuario` (`usuario`),
ADD UNIQUE KEY `correo` (`correo`);

-- 
-- Indices de la tabla `empleado`
-- 

ALTER TABLE `empleado`
ADD PRIMARY KEY (`id`),
ADD UNIQUE KEY `usuario` (`usuario`),
ADD UNIQUE KEY `correo` (`correo`);

-- 
-- Indices de la tabla `inmueble`
-- 

ALTER TABLE `inmueble`
ADD PRIMARY KEY (`id`),
ADD KEY `fk_inmueble_propietario` (`idPropietario`);

-- 
-- Indices de la tabla `propietario`
-- 

ALTER TABLE `propietario`
ADD PRIMARY KEY (`id`),
ADD UNIQUE KEY `usuario` (`usuario`),
ADD UNIQUE KEY `correo` (`correo`);

-- 
-- AUTO_INCREMENT de las tablas volcadas
-- 

-- 
-- AUTO_INCREMENT de la tabla `administrador`
-- 

ALTER TABLE `administrador`
MODIFY `id` int(255) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=3;

-- 
-- AUTO_INCREMENT de la tabla `empleado`
-- 

ALTER TABLE `empleado`
MODIFY `id` int(255) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=5;

```

```

-- AUTO_INCREMENT de la tabla `inmueble`
--
ALTER TABLE `inmueble`
  MODIFY `id` int(4) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=20;

--
-- AUTO_INCREMENT de la tabla `propietario`
--
ALTER TABLE `propietario`
  MODIFY `id` int(255) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=21;

--
-- Restricciones para tablas volcadas
--

--
-- Filtros para la tabla `inmueble`
--
ALTER TABLE `inmueble`
  ADD CONSTRAINT `fk_inmueble_propietario` FOREIGN KEY (`idPropietario`) REFERENCES `propietario` (`id`) ON
DELETE NO ACTION ON UPDATE NO ACTION;
COMMIT;

/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
```