

Nombre	Añadir Club
Resumen	Se le piden todos los atributos de club al usuario, finalmente se agrega a la lista de club que se encuentra en la clase Investor.
Entradas	Los atributos de club.
Resultado	Si el usuario ingresa todos los datos de forma correcta se agregará el club a la lista de clubes de la clase Investor, si llega a tener un error al ingresar estos datos aparecerá un mensaje de que no se pudo agregar el club.

Nombre	Añadir Dueño
Resumen	Se le piden los atributos de dueño al usuario, además de estos se solicita que ponga el nombre del club en el que desea agregar ese dueño.
Entradas	Atributos de dueño y el nombre del club.
Resultado	Si el usuario ingresa todos los datos de forma correcta el dueño queda añadido, si el club al que desea agregarlo no existe se mostrará un mensaje de error, si en la lista del club al que se desea añadir ese dueño ya existe un dueño con el mismo número ID aparecerá un mensaje de error, si introduce los datos mal aparecerá un mensaje de error.

Nombre	Añadir Mascota
Resumen	Se le piden los atributos de mascota al usuario, además de estos se solicita que ponga el nombre del club al que pertenece el dueño al que se la va a agregar la mascota y finalmente se pide el nombre del dueño al que se le desea agregar la mascota.
Entradas	Atributos de mascota, el nombre del club y el nombre del dueño
Resultado	Si el usuario ingresa todos los datos de forma correcta la mascota queda añadida, si el club o el dueño al que se desea agregar la mascota no existe se mostrará un mensaje de error, si en la lista de dueño al que se desea añadir esa mascota ya existe una mascota con el mismo número nombre aparecerá un mensaje de error, si introduce los datos mal aparecerá un mensaje de error.

Nombre	Ordenar Listas
Resumen	Se ordena la lista que se solicite (lista de clubes, dueños o mascotas), a partir del criterio del que se desea ordenar.
Entradas	El criterio por el que se desea ordenar
Resultado	Si se ingresa un criterio valido se muestra en pantalla la lista ordenada por el criterio deseado y se muestra de menor a mayor, ya sea alfabéticamente o numéricamente.

Nombre	Búsqueda
Resumen	Se podrá buscar por cualquier criterio atributo de cualquiera de las clases (por ejemplo, el atributo nombre), se realizará la búsqueda por el método tradicional y por el método de búsqueda binaria.
Entradas	El criterio o atributo por el que se desea buscar.
Resultado	Se muestra en pantalla el tiempo que tardo cada uno de los métodos en realizar la búsqueda.

Nombre	Eliminación
Resumen	Se eliminará el objeto que se desee ya sea club, dueño o mascota
Entradas	Id o nombre del club si se quiere eliminar un club, nombre completo o id si se quiere eliminar un dueño y para mascotas nombres o id
Resultado	Se muestra en pantalla el tiempo que tardo cada uno de los métodos en realizar la búsqueda.

Nombre	Guardar objetos
Resumen	Si se trata de los clubes se guardarán en un archivo texto, por otro lado, dueños y mascotas se guardan en archivos serializables.
Entradas	ninguna
Resultado	Se crea un archivo texto con la información de los clubes y un archivo serializable que contiene a os dueños estos contienen a las mascotas.

Nombre	Leer objetos
Resumen	Si se trata de los clubes se deberá hacer Split y agregarlos manualmente mediante el addClubs, mientras que con los serializables solo se utiliza el object input para traer de nuevo al sistema el ArrayList
Entradas	ninguna
Resultado	Se reingresan los objetos al programa

Requerimiento	Métodos	Clases
Añadir club	<code>public String addClub (int iD, String clubName, String creationDate, String typesOfPets)</code>	Investor

Requerimiento	Métodos	Clases
Añadir dueño	<code>public String addOwners (String clubName, int cC, String name, String lastName, String birthDay,String petPreference)</code> <code>public String addOwner (int Cc, String name, String lastName, String birthDay, String petPreference)</code>	Investor Clubs

Requerimiento	Métodos	Clases
Añadir mascota	<code>public String addPet(String nc, int cc, int pID, String name, String birthDay, char gender, char type)</code> <code>public Owners searchOwnerId(String nc, int cc)</code> <code>public String addPet(int pID, String name2, String birthDay2, char gender, char type)</code>	Investor Clubs Owners

Requerimiento	Métodos	Clases
Ordenar listas	<pre> public void organizeOwnerLastName(String nc) public Clubs searchClubName(String nc) public void organizeOwnerLastName(String nc) public void bubbleOwnerLastName() public void organizeOwnerName(String nc) public void insercionOwnerName() public void organizeOwnerNumberPets(String nc) public void bubbleOwnerNumberOfPets() public void organizePetName(String nc, int id) public Owners searchOwner(int Cc) public void bubbleNamePet() public void organizePetID(String nc, int id) public void bubblePID() </pre>	Investor Clubs Owners

Requerimiento	Métodos	Clases
Búsqueda	<pre> public void bubbleClubName() public String searchClubNameComparation(String nc) public void bubbleClubID() public String searchClubIDComparation(int id) public void organizeOwnerLastName(String nc) public String searchOwnerLastNameComparation(String nc, String lno) public void organizeOwnerName(String nc) public String searchOwnerNameComparation(String nc, String no) public String searchOwnerNameComparation(String no) public String searchOwnerLastNameComparation(String lno) public void organizePetName(String nc, int id) public String searchPetName(String nc, int cc, String pn) public String searchPetName(String pn) public Clubs searchClubName(String nc) public Owners searchOwnerId(nc,cc) public String searchPetID(String nc, int cc, int pid) </pre>	Investor Club Owner Pet

```
public String search(int pid)
```

