¿Qué debe hacer el programa? Haga una lista de las opciones que brindará al usuario en la interacción con su sistema. Por ejemplo. “Asignar Mascota a Familia”…

**Asignar macotas, ingresar datos de las 3 primeras familias, recibir can, permitir ingresar datos del perro, asignar can a la familia, sino se ubica el perro se habilita un espacio.**

- ¿Qué clases necesitará para dar solución a la situación planteada?

**Familias, perros y principal**

- ¿Qué propiedades y métodos tendrá cada clase?

**Familias niños menores de 10 años. -> hijos edad de hijos**

**Familias con niños mayores de 10 años. -> Hijos edad de hijos**

**Familias sin niños -> hijos edad de hijos**

**Perros-> tamaño, raza, edad, estado de salud, color, nombre**

- ¿Qué tipo deben tener las propiedades y métodos de cada clase?

**Familias niños menores de 10 años. -> hijos(string) edad de hijos(int)**

**Familias niños mayores de 10 años. -> hijos(string) edad de hijos(int)**

**Familias sin hijos -> hijos(string) edad de hijos(int)**

**Perros-> tamaño(string), raza(string), edad(int), estado de salud(float), color(string), nombre(string)**

- ¿Cuáles deben ser los modificadores de visibilidad de los miembros en cada clase?

**Agarrar perro y guardarlo, atributos de cada perro, si la familia tiene hijos o no y si son mayores o menores de 10 años.**

- ¿Qué parámetros serán requeridos por los métodos en sus clases?

**Familias hijos menores a 10 solo pueden tener perros pequeños y labradores**

**Familias hijos mayores a 10 pueden tener pequeños y medianos y exceptuando razas peligrosas.**

**Familias sin hijos pueden tener a cualquier perro**

- ¿Cómo proveerá de valores iniciales a sus objetos?

**En cada clase se le va asignar un valor a los atributos del perro y de la familia.**

¿Qué valores iniciales les asignará?

**Familias niños menores de 10 años. -> hijos(“”) edad de hijos(0)**

**Familias niños mayores de 10 años. -> hijos(“”) edad de hijos(0)**

**Familias sin hijos -> hijos(“”) edad de hijos(0)**

**Perros-> tamaño(“”), raza(“”), edad(0), estado de salud(0), color(“”), nombre(“”)**