```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

typedef struct _puppy{
    int age;
    char* name;
} puppy;

puppy *alloc_puppy()
{
    puppy *p = (puppy *)malloc(sizeof(puppy));
    p->age = 0;
    p->name = (char *)malloc(sizeof(char)*5);
    strcpy(p->name, “Fido”);
    return p;
}

puppy *alloc_init_puppy(int age, char *name)
{
    puppy *p = (puppy *)malloc(sizeof(puppy));
    p->age = age;
    p->name = strdup(name);
    return p;
}

void puppy_setName(puppy *this, char *name)
{   this -> name = name; }

void puppy_setAge(puppy *this, int age)
{   this->age = age; }

char *puppy_getName(puppy *this)
{   return this->name; }

int puppy_getAge(puppy *this)
{   return this-> age;  }
```

```
public class Puppy{
    private int age;
    public String name;


    public Puppy(){
        this.age = 0;
        this.name = “Fido”;
    }



    public Puppy(int a, String n)
    {
        this.age = a;
        this.name = n;
    }



    public void setName(String n)
    {  this.name = n; }

    public void setAge(int age)
    { this.age = age; }

    public String getName()
    { return this.name; } // return name;

    public int getAge()
    { return this.age;}   // return age;
```

```
int main(int argc, char* argv[])
{
    int i;
    for(i=0;i<argc;i++)
        printf(“%d: %s\n”,i,argv[i]);


    puppy *myPuppy = alloc_puppy();
    puppy_setName(myPuppy, “Tommy”);
    puppy_setAge(myPuppy, 2);
    int myAge = puppy_getAge(myPuppy);

    printf(“Value of myAge variable : %d \n”,
                myAge);

    free (myPuppy);

    myPuppy = alloc_init_puppy(5, “Fluffy”);
    printf(“MyPuppy: %d, %s\n”,
        puppy_getAge(myPuppy),
        puppy_getName(myPuppy));

    printf(“Name: %s\n”,myPuppy->name);
    return 0;
}
```

To compile:

```
gcc puppy.c –o puppyProgram
```

to run:

```
./puppyProgram
```

```
    public static void main(String [] args)
    {
        int i;
        for(i=0;i<args.length;i++)
            System.out.println
                (“” + i+ “: ” + args[i] + “\n”);

    Puppy myPuppy = new Puppy();
    myPuppy.setName(“Tommy”);
    myPuppy.setAge(2);
    int myAge = myPuppy.getAge();

    System.out.println(“Value of myAge”
            + “variable : ” + myAge);

    // no need to delete in Java – it will reclaim
    // “garbage collection”
    myPuppy = new Puppy(5, “Fluffy”);
    System.out.println( “MyPuppy: ” +
        myPuppy.getAge() + “, ” +
        myPuppy.getName() + “\n”);

    System.out.println(“Name: ” + myPuppy.name
                      + “\n”);

    } // end of main
} // end of class
```

To compile:

```
javac Puppy.java
```

To run:

```
java Puppy
```

Use IntelliJ to compile, run, and debug!!!