#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; }

using System;

public class Puppy{

 private uint age;

 public string name;

 public Puppy(){

 age = 0;

 name = “Fido”;

 }

 public Puppy(uint a, string n)

 {

 age = a;

 name = n;

 }

 public void setName(string n)

 { name = n; }

 public void setAge(uint a)

 { age = a; }

 public String getName()

 { return name; } // return name;

 public uint getAge()

 { return age;} // return age;

} // end of class Puppy

int main(int argc, char\* argv[])

{

 unsigned 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

using System;

public class TestPuppy {

 public static void Main(string[] args)

 {

 uint i;

 for(i=0;i<args.Length;i++)

 Console.WriteLine

 (“” + i+ “: ” + args[i]);

 Puppy myPuppy = new Puppy();

 myPuppy.setName(“Tommy”);

 myPuppy.setAge(2);

 uint myAge = myPuppy.getAge();

 Console.WriteLine(“Value of myAge”

 + “variable : ” + myAge);

 // do not delete in C# – it will reclaim

 // “garbage collection”

 myPuppy = new Puppy(5, “Fluffy”);

 Console.WriteLine ( “MyPuppy: ” +

 myPuppy.getAge() + “, ” +

 myPuppy.getName());

 Console.WriteLine(“Name: ” +

 myPuppy.name);

 } // end of Main

} // end of class Puppy

To compile:

mcs TestPuppy.cs Puppy.cs

To run:

TestPuppy.exe