

# Protocolos en Objective-C

# Interfaces

Separan el protocolo ( como me comunico con ) y el mecanismo ( la implementación del comportamiento )

## Animal Interface

```
public interface Animal
{
    public void locomotion();
    public void eat();
}
```

## Class Shark

```
public class Shark implements Animal
{
    public void locomotion()
    {
        System.out.println("I swim.");
    }
    public void eat()
    {
        System.out.println("I hunt for seals.");
    }
}
```

## Class Dog

```
public class Dog implements Animal
{
    public void locomotion()
    {
        System.out.println("I run on four legs.");
    }
    public void eat()
    {
        System.out.println("I eat kibble.");
    }
}
```

```
public class AnimalTest
{
    public static void main(String[] arg)
    {
        Shark shark = new Shark();
        shark.locomotion();
        shark.eat();
        Dog dog = new dog();
        dog.locomotion();
        dog.eat();
    }
}
```

# Protocolos en Objective-c

```
// Definicion ( @protocol )  
@protocol Animal <NSObject>
```

```
-(void) locomotion;  
-(void) eat;
```

```
@end
```

```
int main (int argc, const char * argv[])  
{  
    @autoreleasepool {  
        Dog *dog = [[[Dog alloc] init] autorelease];  
        [dog locomotion];  
  
        id<Animal> animal = [[[Shark alloc] init] autorelease];  
        [animal eat];  
    }  
    return 0;  
}
```

```
@implementation Shark
```

```
-(void) locomotion  
{  
    NSLog(@"I swim");  
}
```

```
-(void) eat  
{  
    NSLog(@"I hunt for seals");  
}  
@end
```

```
@interface Dog : NSObject<Animal>  
@end
```

```
@implementation Dog
```

```
-(void) locomotion  
{  
    NSLog(@"I run on four legs");  
}
```

```
-(void) eat  
{  
    NSLog(@"I eat kibble");  
}  
@end
```

# Propiedades