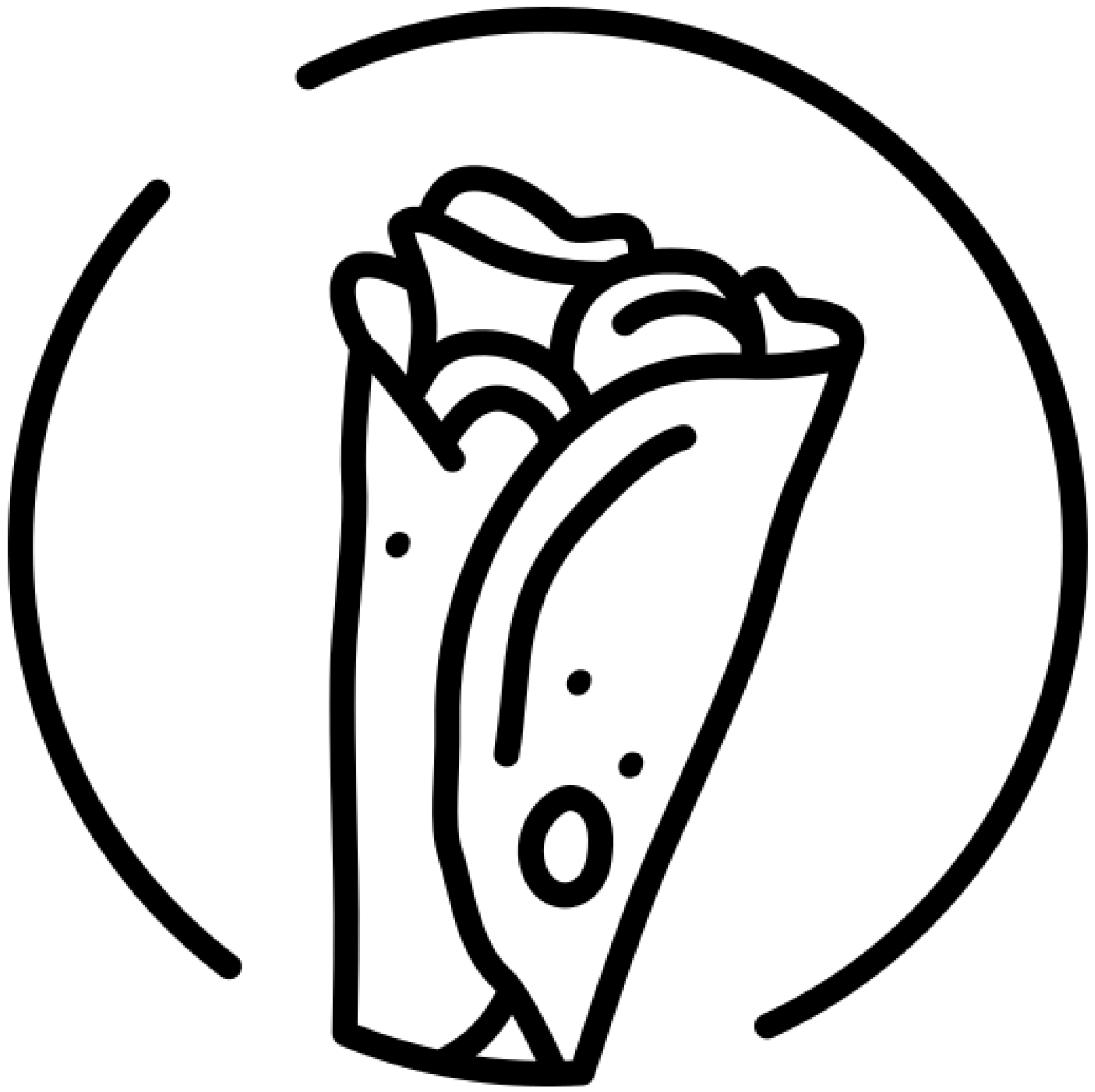
Repository Wrapper Pattern with C#



romain-od

www.code-review.tech

THE CONTEXT

Situation:
Need to retrieve all owners and certain accounts in a controller

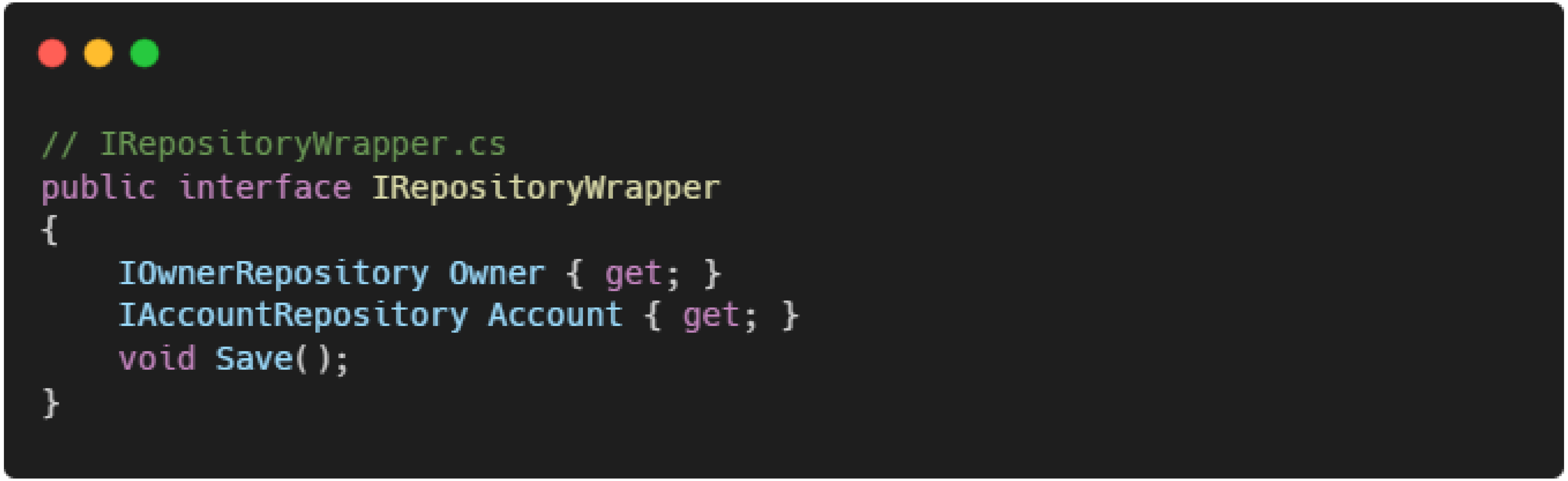
Solution:
Instantiate OwnerRepository and AccountRepository classes and use FindAll and FindByCondition methods

Problem:
Cumbersome if needing logic from five or more classes

Simplification:
Create a wrapper around repository user classes, add to IOC, and inject into controller's constructor

Benefit:
Easily call any repository class needed with the wrapper instance.

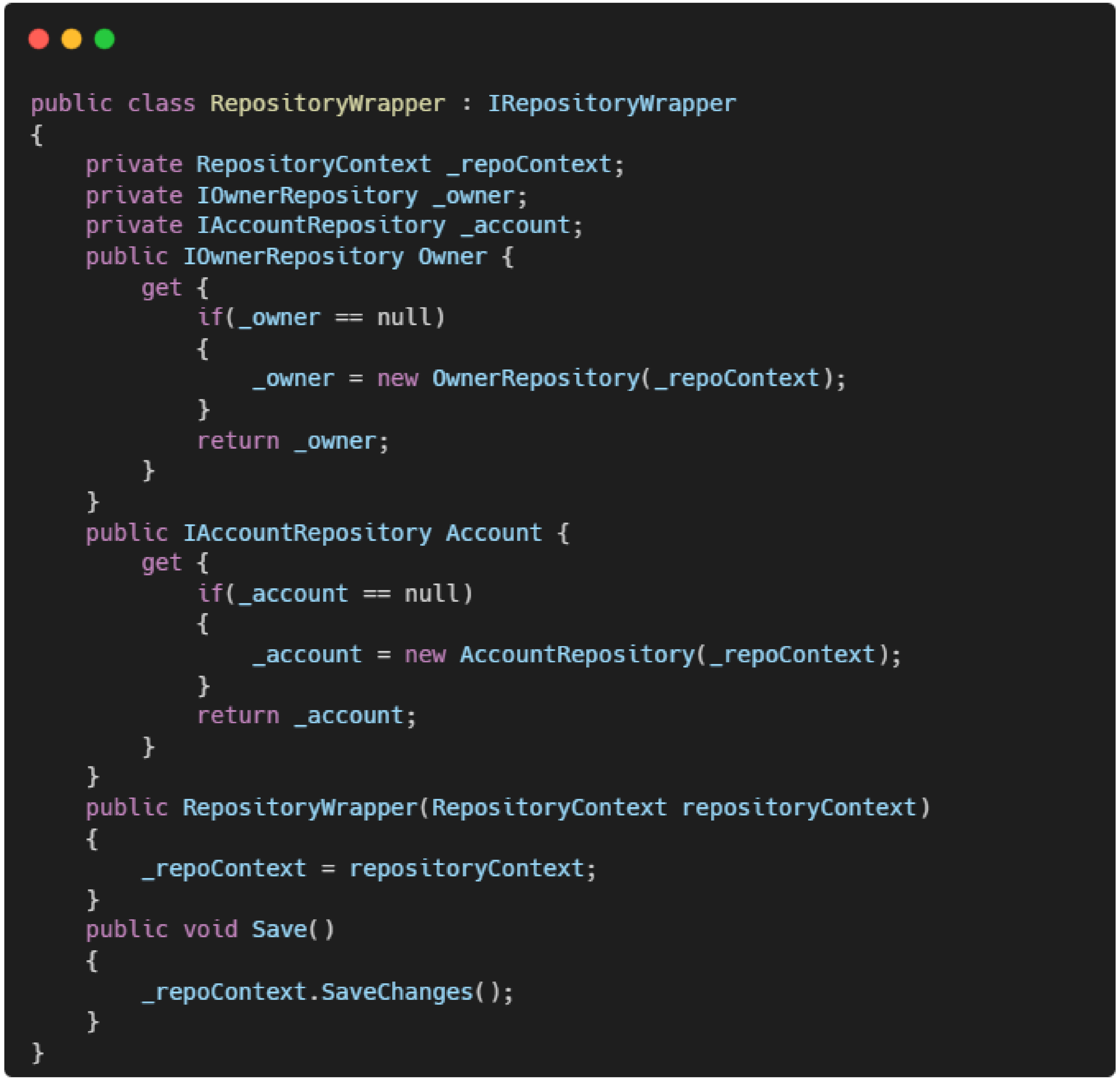
FIRST CREATE THE INTERFACE



```
// IRepositoryWrapper.cs
public interface IRepositoryWrapper
{
    IOwnerRepository Owner { get; }
    IAccountRepository Account { get; }
    void Save();
}
```

romain-od

www.code-review.tech



```
public class RepositoryWrapper : IRepositoryWrapper
{
    private RepositoryContext _repoContext;
    private IOwnerRepository _owner;
    private IAccountRepository _account;
    public IOwnerRepository Owner {
        get {
            if(_owner == null)
            {
                _owner = new OwnerRepository(_repoContext);
            }
            return _owner;
        }
    }
    public IAccountRepository Account {
        get {
            if(_account == null)
            {
                _account = new AccountRepository(_repoContext);
            }
            return _account;
        }
    }
    public RepositoryWrapper(RepositoryContext repositoryContext)
    {
        _repoContext = repositoryContext;
    }
    public void Save()
    {
        _repoContext.SaveChanges();
    }
}
```

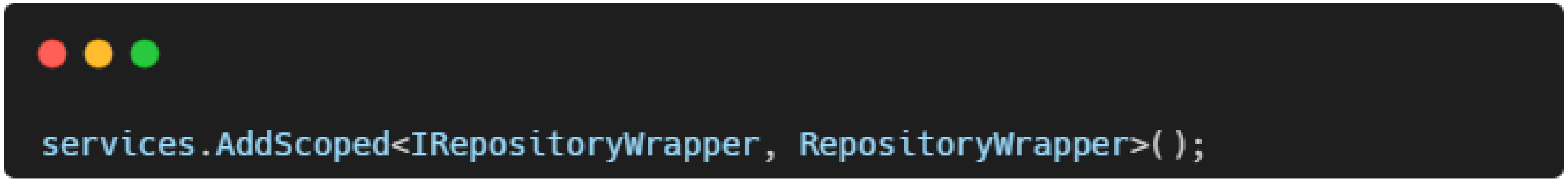
NEXT, INTRODUCING A NEW CLASS

- Creating properties to expose concrete repositories.
- Adding a Save() method to the class to be used after modifications are complete.
- Modify multiple objects in one method and then call the Save() method once.

romain-od

www.code-review.tech

FINALLY THE IOC



```
services.AddScoped<IRepositoryWrapper, RepositoryWrapper>();
```

Now you can call the repository anywhere you need it with DI

romain-od

www.code-review.tech

THANKS FOR READING

SHARE YOUR THOUGHTS

romain-od

www.code-review.tech