



Av Erik
"Zkall"
Svensson
@esdev.se



C# Interface

Vad är ett interface?

Ett interface är som en mall för hur en class ska se ut. Det är tomma funktioner som aldrig implementeras utan bara deklareras. Man kan även kalla ett interface för ett kontrakt mellan olika program lager när man använder sig av dependency injection. För att förenkla det hela, interface är en deklaration av hur en class ska se ut och består av tomta påståenden.

Hur ser ett interface ut?

```
public interface IMenyBusinessLogicLayer
{
    int GetDokumentCount(string username);
    int GetNyheterCount(string username);
    int GetChattCount(string username);
    void GetForumCount(string username);
    void MarkAllChattasRead(string username);
    void MarkAllNyheterasRead(string username);
    void MarkAllDokumentasRead(string username);
    void List<NyheterVM> GetNyheter();
    void AddNyheter(NyheterVM nyheten);
}
```

Hur ser en implementerad class ut från ett interface?

```
public class MenyBusinessLogicLayer : IMenyBusinessLogicLayer
{
    public int GetDokumentCount(string username)
    {
        var dokumentcount = db.Dokumenter.ToList().Count;
        var lästadokument = db.LästaDokument.Where(x => x.User.user.UserName == username).ToList().Count;

        return dokumentcount - lästadokument;
    }
    public int GetNyheterCount(string username)
    {
        var nyhetercount = db.Nyheter.ToList().Count;
        var lästanyhetercount = db.LästaNyheter.Where(x => x.User.user.UserName == username).ToList().Count;
        return nyhetercount - lästanyhetercount;
    }

    Etc...
}
```

Slutsats

Du kan skapa interface från en class genom att högerklicka refactor, extract interface. Då skapas deklarations interfacet för din class och du kan lita dig tillbaka som det mesta skött sig själv.

För dependency injection är interfacet ett måste. Mer om det i dependency injection dokumentet!