|  |  |
| --- | --- |
| **Øvelse** | LINQShop |
| **Projekt** | LINQShop |
| **Formål** | Brug af LINQ til at implementere beregninger og funktionalitet |
| **Beskrivelse** | Den givne Visual Studio solution rummer fra starten følgende klasser, der skal repræsentere data om en butik:   * **Produkt og Kunde**: Simple domæne-klasser, uden referencer til andre klasser * **Ordre**: Mere kompleks domæne-klasse, idet et **Ordre**-objekt refererer direkte til et **Kunde**-objekt, samt til flere produkter via et *Id*. * **DataModel**: Rummer alle data for butikken, dvs. alle **Kunde**-, **Produkt**- og **Ordre**-objekter. Denne klasse rummer også flere komplekse properties og metoder. |
| **Trin** | 1. Download og unzip (VIGTIGT) filen **LINQShop.zip** fra Moodle. 2. Åbn **LINQShop** i Visual Studio. Start med at få et overblik over programmet. De to klasser **Produkt** og **Kunde** er komplette, mens der skal implementeres flere elementer i **Ordre** og **DataModel**. 3. Fortsæt til klassen **Ordre**. Kommentarerne skulle gerne indikere intentionen med properties og metoder. Dog er property **TotalPris** ikke gjort færdig, det skal du gøre 😊. Og prøv meget gerne at bruge **LINQ** til at implementere denne property! Husk at prøve at teste din kode i **Program.cs**. 4. Fortsæt videre til klassen **DataModel**. Her er der adskillige properties og metoder der ikke er gjort færdige (se efter **// TODO** i klassen). Igen skulle kommentarerne gerne beskrive intentionen, så dit job er igen at få gjort disse elementer færdige, inklusive at afprøve det du implementerer. Langt de fleste elementer burde kunne implementeres med **LINQ**, men hvis det kniber, kan det også være en god tilgang at få det til at virke f.eks. med brug af løkker og *if*-statements, og så efterfølgende prøve at konvertere til **LINQ**. |