

Bageriet 1.0

Jag har efter denna vecka lärt mig att programmera enligt MVC-metoden.

Genom att jobba på detta sättet separerar man uppgifter i programmet, vilket gör att det blir lättare att underhålla och bygga programmet.

MVC innebär i kort:

Model: innehåller data som programmet jobbar med, alla klasser, databaser

View: via denna visas informationen till användaren

Controller: styr allting, tar userinput/url, tar data från model och skickar info till View

Jag har efter gått genom kursmaterialet kommit fram till följande slutsats:

Model:

Class Cake, här lägger jag alla attribut.

Class MockCake, här initiera jag Cake och lägger det i en lista

Interface ICake, här skapa jag kontrakt med MockCake och det är med denna jag jobbar mot Controller.

Via Controller skickar jag sedan informationen till View som sedan kan visas med hjälp av @model i cshtml filen.

Jag vet inte om jag är rätt ute, tar gärna emot feedback.

I denna inlämningsuppgift har jag valt att bygga en superklass, product. Genom att göra så tänker jag mig det blir lätt att lista alla framtida produkter (tårta, bröd, bakelser etc) mha List<Product>.

Har ingen aningen om detta är ett bra sätt att jobba då pluralsight kursen verka löst det genom att ha alla klasser vart för sig och infört kategori istället.

Feedback önskas, då jag få ändra till bageri 2.0 om detta inte funkar bra 😊.

Överlag tycker jag att det har varit ganska klurigt, MVC är helt nytt för mig så har fått titta genom kursmaterial flera gånger och sökt i youtube för att få en så bra förståelse som möjligt.

Dessutom tyckte jag att föreläsaren går för snabbt framåt, vissa saker är oklart

- Vad är interface?
- Vad gör IEnumerable?

Har försökt hitta svar själv via youtube och google, gick väl sådär.

Namn: Fon To

Github: <https://github.com/fonto1/Bakery.git>