1/3/2023

Abdel-Rahman Gamal Ahmed

no campany

Design pattern

Data access pattern



Table of Contents

[1- Topic: 2](#_Toc123759404)

[2-simple app for test: 2](#_Toc123759405)

[4- Repository pattern : 6](#_Toc123759406)

# 1- Topic:

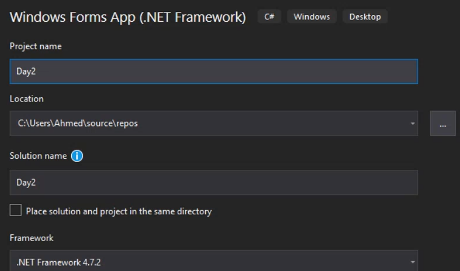
* هناخد النهاردة ال repository pattern ودة واحد من اهم ال pattern الى هنشفها وبنصنفه على انه data access pattern .
* لو انت هتشتغل backend من ضمن ال layer الى هتبنيها هى ال repository layer .
* وال المرة الجايه هناخد حاجة اسمها dependence injection principle
* والمرة الجايه برضو هناخد ال structure pattern كمان على السريع كدة .

# 2-simple app for test:

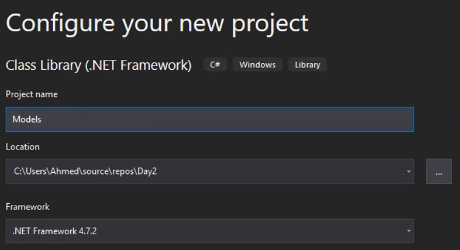
ف الاول كدة هنعمل على السريع مشروع بسيط وكنا علمناه قبل كدة لمجرد ان نطبق عليه مش اكتر .

* واحنا هنعمل اكتر من layer الى كنا بنعملهم زمان
* Business layer ودى الى هيكون فيها ال model بتاعتى وهيكون ايمها model
* وlayer هيكون هنعمل فيها map للmodel بتاعتنا ف ال database . يعنى هتكون layer فيها شغل ال EF
* UI layer .
* بعد الكام layer دول والى اصلا عملنها قبل كدةومفيش فيهم جديد هنعمل ال repository layer .

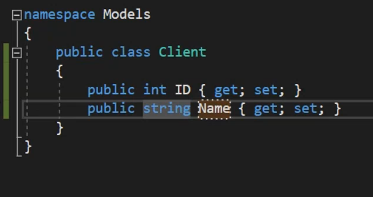
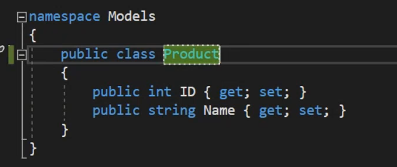
1. اول مشروع هنعمله هو بتاع ال UI : واسمه هيكون day2 وهنرج ليه بعد شويه



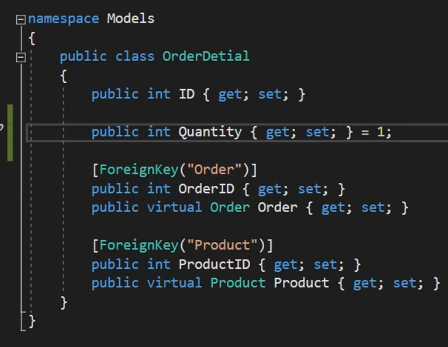
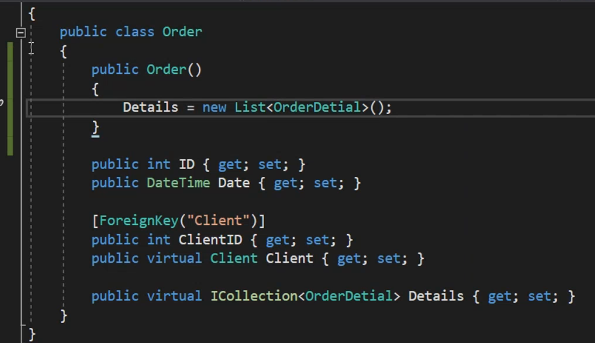
1. تانى حاجة هنعملها business layer : وطبعا دة من نوع class library . وهنعمل فيه order system صغير



* اول مودل هو ال product والتانى هو ال client .

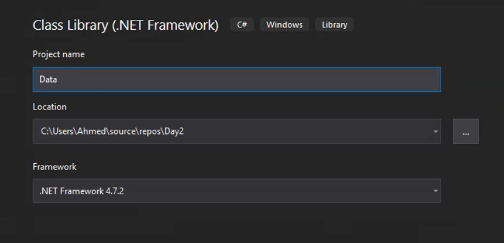


* تالت كلاس هو ال order وهيكون ف علاقه 1:m مع ال client وكمان علاقه m:m مع ال product فهنعمل bridge اسمهorderDetails ودة هيكون فيه كل سطر ف ال order عباره عن ايه او وصف كل سطر ف الorder .

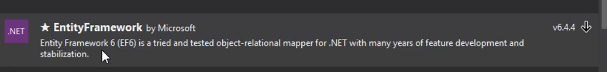


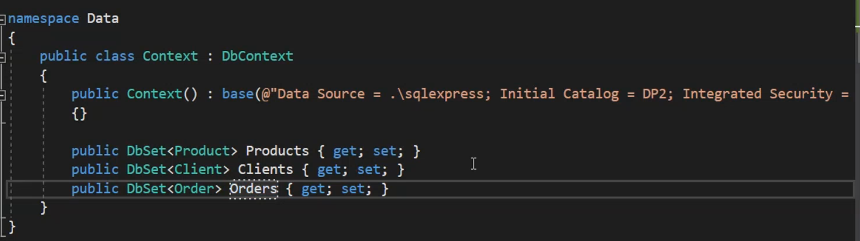
* مش محتاح اقلك ان الكلاس دى لمجرد الشرح .

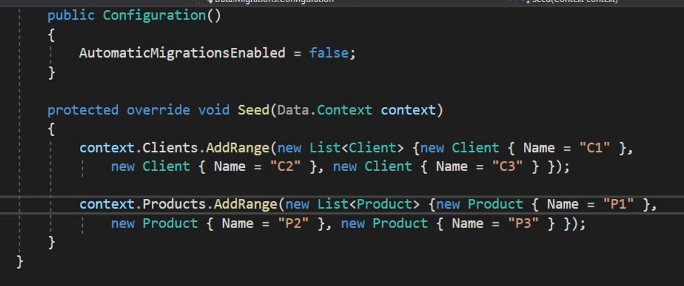
1. تالت layer الى هيكون فيها موضع الmap بين ال models والdatabase . وهو برضو class library :



1. وهنحتاج نحمل ال EF packages .

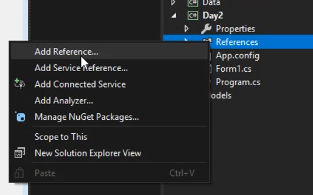


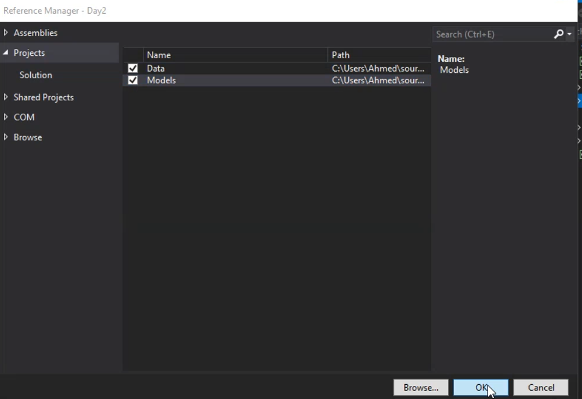
* ف دلوقتى هنعمل ال DB context class الى هيكون فيه ال connectionString وكمان ال class الى هعملها mapping
* دة ال connectionString وبنحطه ف كونستراكتور بتاع ال parent .
* ودول الكلاس بتوعنا ال هنعملهم map ف الداتا بيز
* وهنحط بس داتا ف seed method عشان واحنا بنعمل migration تتخزف ال DB . لو ناسى ال seed method هتكون ف migration class الى هيظهرك .

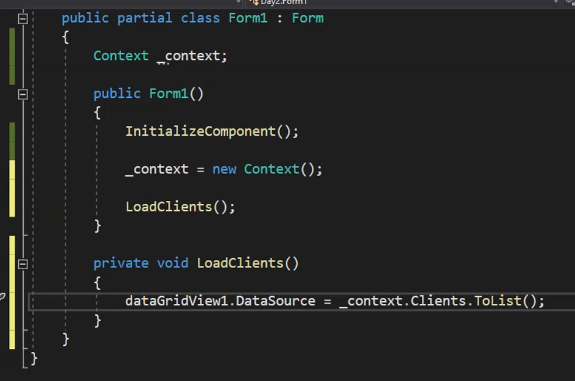


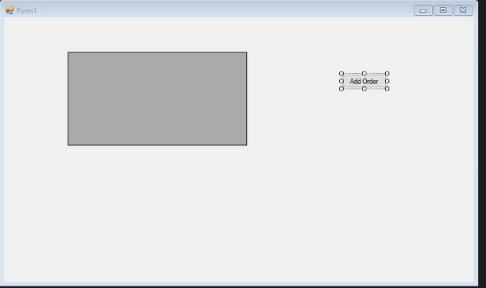
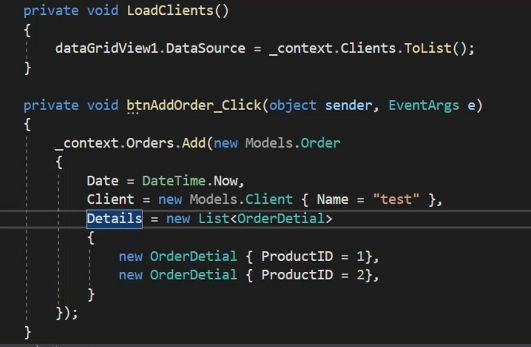
1. تانى حاجة هنعملها enable migration و add migration و update database .
2. كدة خلصنا الlayer بتاعه ال model وكمان ال layer بتاعه الmapping ف كدة جاهزين عشان نحط داتا ف الداتابيز .
3. هنرجع نعمل شويه شغل بسيط ف ال layer بتاعه ال UI الحجات الى هنعملها لمجرد التست الشغل ولفهم المشكله الى هتخلينا نعمل ال repository .

* وعشان نشتغل مع ال layer بتاعه الداتا هنعمل add reference ليها ف مشورع ال UI ال اسمه day2



* هحدد انا عاوز اعمل add reference لمين باضبط
* وهنعمل install للpackages بتاع EF طلام هنتعامل مع DB Context .
* بس كدة جاهزين نكتب كود جوة ال ui layer .

1. الكود الى هنعمله بسيط خالص : اول حاجة عملت grid هعرض فيها ال client . وطبعا عملت object form context .

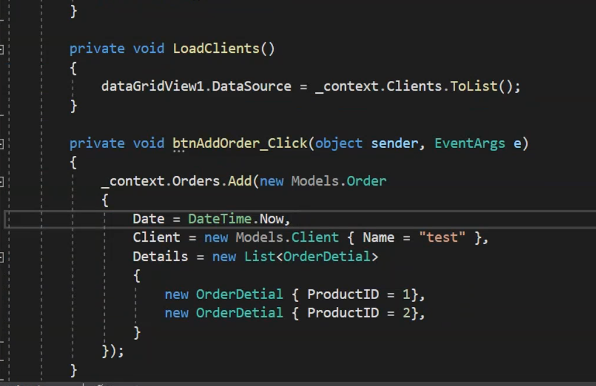
* ودة تانى كود بيخليك لما ادوس على زرار يضيف order والاسكرين الى على اليمين دى بتاعه ال ui .

# 4- Repository pattern:

الكود الى فات احنا اصلا عملناه قبل كدة وانا كتبته بس عشان هنكمل عليه ف يعتبر مفهوش ولا معلومه جديده:

تعاله بقا نتناقش ونشوف ليه هنزود ال access layer data :

* ايه المشكله ف الكود بتاع ال ui ودة اسكرين لجزء منه وقصدى بالمشكله يعنى ايه المبادئ بتاع ال solid الى متحققتش :



1. اول مشكله ممكن تلحزها ان ال ui بيستخدام ال database يعنى بيستخدام concrete object من ال context مباشرتا .

* يعنى ال ui layer كدة بقت tightly couple مع ال layer بتاعه ال data وال model layer .
* موضوع ال tightly couple مع ال model layer هناقشه ف ال mvc pattern . خلينا دلوقتى ف ال tightly couple مع data layer .