ניקח לדוגמא את טבלת user מהדטה ולכל טבלה וטבלה ניצור שכבות אלו

צד שרת c#

הערות:

מתעסק רק עם מלוכלכים!! מחלקת user עם כל המשתנים כולל המלוכלכים- מחלקה זו היא אוטומטית שנוצרת עם יצירת הדטה

מתעסק רק עם מלוכלכים!!

מחלקת converterממיר מלוכלך לנקי כלומר userDTO- כאשר המחלקה הזו היא עם משתנים נקיים ללא virtual כלומר מחלקת bll שולחת בפונקציה למחלקת converter ששם ישנה פונקציית המרה ומחזירה לbll נקי. בחזור מהאנגולר זה ילך לפונקציית המרה מנקי למלוכלך וכך ימשיך הלאה ל-dal שמתעסק רק עם מלוכלכים וכן הלאה לdb

מתעסק רק עם נקיים כלומר בפונקציה במחלקת controller מוחזר פונקציה נקייה מה-bll. תפקידו: מקבל בקשות ומעביר אותן הלאה.

שכבה זו היא לא מתקשרת בפועל היא רק מכינה לנו תבניות של אובייקטים נקיים שאותם נעביר הלוך וחזור לאנגולר לצד לקוח.

כל המחלקות שבתיקיה זו (בעצם לכל טבלה יש את המחלקה שלה )מקבילות לכל המחלקות שב- DTO שב-C# בעצם גם ב- modelsוגם ב-DTO אלה משתנים נקיים.

בתיקיה זו יש לכל טבלה את מחלקת ה- service שלה שזה בעצם מקביל למחלקת-controller שנמצא בשכבת API שב-C#.

המחלקות שבתוך השכבות:

השכבות:

db

User

userdal

dal

userConverter

userbll

bll

api

userController

userDTO

DTO

user. service.ts

services

User.modle.ts

models

צד לקוח אנגולר