Opgave: Model Klasser Med Begrænsninger (constraints)

I bagerbutikken 'Ø-Bageren' har de brug for et IT-system til at holde styr på brød og kager.

a. Du skal lave et nyt projekt (solution) Class Library fx 'BakeryLib'

Model Klasser

b. Du skal lave en folder 'model', hvor du opretter en abstrakt klasse **'BakeryItem'**, der har følgende attributter – dvs. instans felter (IKKE PROPERTIES):
   i. name (string)
   ii. id (int)
   iii. price (double)
   iv. isCake (bool)

c. Du skal lave to konstruktører
   i. En default konstruktør – uden parametre.
   ii. En konstruktør med parametre til at initialisere name, id, price og isCake.

d. Du skal lave properties til name, id, price og isCake.

e. Du skal lave en metode ToString, der returnere en tekst med værdierne af attributterne/Property'ene.

f. Du skal lave endnu en klasse **'Bread'** der arver fra BakeryItem og som desuden har følgende attributter:
   i. Weight (int)
   ii. FullGrain (bool)

g. Du skal lave properties til de to instans felter, to konstruktører (default og en der sætter alle properties), Samt en ToString()

h. Du skal lave endnu en klasse **'Cake'** der arver fra BakeryItem og som desuden har følgende attributter:
   i. KindOfCake (string)

   alternativt typen er en enum

i. Du skal lave properties til de instans feltet, to konstruktører (default og en der sætter alle properties), Samt en ToString()

Repository Klasse

a. Du skal lave en folder 'repository', hvor du opretter en klasse 'BakeryRepository', den skal have en liste af BakeryItems, samt have følgende metoder:
   i. GetAll, det returnerer en liste af bakeryItems
   ii. GetAllBread(), der returnerer en liste af Bread
   iii. GetAllCake(), der returnerer en liste af Cake
   iv. Add(BakeryItem item), der tilføjer et item til repoistoriet
   metoden skal autogenerere et unikt id til item objektet, objektet med det nye id returneres
   v. Delete(int id), finder og sletter et bakeryItem objekt med id, det objekt der slettes skal returneres. Hvis der ikke findes et objekt kastes en passende exception

Ekstra: Update(int id, BakeryItem item) finder og updaterer et bakery objekt med angivet id, det opdaterede objekt der returneres. Hvis der ikke findes et objekt kastes en passende exception

b. Du skal lave en konstruktør
   En default konstruktør – uden parametre.

c. Du skal lave en metode ToString, der samler alle bakeryItems i en tekst streng.

d. Husk at 'build'e dit Library

Program

e. Du skal lave et nyt projekt .Net Console App fx 'BakeryApp'

f. Du skal under dependencies lave en reference til dit BakeryLib

g. Du skal sætte dit nye projekt til at være start-up (højre klik på projekt og vælg 'set startup')

h. Lav to bread objekter og to cake objekter og udskriv dem på skærmen, samt lav et objekt af dit BakeryRepository og prøv de fem metoder, hvor du skriver resultater på skærmen.

Lav begrænsninger på dine modelklasser

Du skal nu lave begrænsninger i de værdier de forskellige attributter kan få. Altså refakturerer dine model klasser. Dvs. du skal ind og ændre set-metoderne i property'ene

i. name skal have mindst 2 tegn og må ikke være null

j. price skal være mere end 0,00 kr. (altså ikke negativ)

k. weight skal være mindst 50 g

l. KindOfCake skal være 'flødeskumskage', 'tørkage', 'wienerbrød' eller 'sandkage'

Hvis de IKKE overholde kravene skal de kaste en ArgumentException eller ArgumentNullException.

m. Prøv i programmet at sætte nogle værdier der ikke er tilladt, benyt try-catch til at fange fejlen og udskriv en fejlbesked.