Eksamen A

I fagene

Programmering og Teknik

2020, juni, kl. 9.00 - 14.00

Alle skriftlige materialer, pc'er, bærbare computere og internet ressourcer er tilladt til eksamen.

Mobiltelefoner og kommunikation med andre personer (inkl. chatbots), bortset fra kommunikation med eksaminator og censor, er forbudt.

Du må ikke gemme dine løsninger på eksterne netværksdrev/værter som GitHub, Facebook o. lign. Overtrædelse af dette vil medføre bortvisning fra eksamen og passende sanktioner vil senere blive tildelt både til afsender/uploader og modtager.

Ved eksamens afslutning skal du lægge din løsning op i Wiseflow.

Eksamen varer 4 timer og efterfølges af 1 times evaluering. I disse tidsrum kan du kun forlade eksamenslokalet med henblik på toilet besøg. Rygning er ikke tilladt.

Eksamenssættet består af 8 opgaver.

Ud over disse opgaver, kan du kan blive bedt at besvare nogle ekstra spørgsmål omkring dine svar og eventuelle andre valg.

Kontroller at dette sæt indeholder 3 sider inklusive forsiden.

# Introduktion

Du skal programmere et system, der kan indsamle data til statistik om Corona virus og sygdommen Covid-19.

Der indsamles data om forekomsten af Corona virus og Covid-19 i husholdninger.

Udover at programmere dette system, er der også teoretiske spørgsmål undervejs, hvor du bliver bedt om at redegøre for overvejelser og nedskrive forklaringer. Dette skal du skrive i et dokument, som uploades til Wiseflow sammen med dine programmer i en zip fil.

# Opgave 1 Class Library

Du skal lave et projekt af typen Class Library (.NET core).

Dit projekt skal indeholde en klasse, *CovidRecord*, der har følgende properties

* *Id*, et hel-tal
* *City*, en tekst-streng,
tekstens længde skal være mindst 2 tegn
* *Household*, et hel-tal
Antallet af medlemmer i husholdningen
*household* >= 1
* *Tested*, et hel-tal
Antallet af medlemmer i husholdningen, som er blevet testet for Corona virus.
*Tested* >= 0
* *Sick*, et hel-tal
Antallet af medlemmer i husholdning, som er eller har været syge efter 1. marts.
*Sick* >= 0

Der skal kastes passende *exceptions,* hvis betingelserne ikke overholdes.

# Opgave 2 Unit test

Du skal lave en unit test af din CovidRecord klasse.

Husk at teste constructors.

Din test skal have et godt “Code Coverage”

# Opgave 3 REST service provider

1. Lav en ASP.NET Core Web Applikation projekt “covidservice”.
Tilføj en reference til dit class library.
2. Lav en controller, CovidRecordsController, som tilbyder metoderne:
	1. Public IEnumerable<CovidRecord> GetAllRecords()
	2. Public CovidRecord GetById(int id)
	3. Public IEnumerable<CovidRecord> GetRecordsByCity(string city)
	4. Public int AddRecord(CovidRecord record)
	*Id* skal genereres af controlleren.
	Metoden returnerer det tildelte *Id*.

Din controller skal bruge en *static* list til at opbevare data.
Tilføj mindst 5 objekter til listen.
Hvad betyder det at listen er *static*?

1. Kør din service lokale (*localhost*).
Afprøv din service med en browser og/eller Postman.
Hvilke metoder kan du afprøve fra browseren?
2. Dokumenter din service vha. Swagger.
Fremvis dokumentationen og vis at du kan eksekvere controller-metoderne fra Swagger.
3. Forklar hvordan din REST service anvender HTTP protokollen.

# Opgave 4 Kryptering

Fortæl kort om symmetrisk og asymmetrisk kryptering.

# Opgave 5 Integration test

1. Test én af metoderne i din REST service i Postman.
2. Beskriv kort forskellen mellem unit test og integration test.

# Opgave 6 REST i Azure + CORS

1. Publicer din REST service til Microsoft Azure.
2. Indstil din REST service til at understøtte CORS.
Det kan gøres på flere måder. Du bestemmer selv hvilken måde du vil benytte.

# Opgave 7 Web-applikation, POST

Lav en web-applikation med HTML og Typescript.

Web-applikationen skal gøre det muligt for en bruger at indtaste de relevante data for en *CovidRecord*.
Bemærk at brugeren ikke skal indtaste *Id*. *Id* genereres af REST service.

Web-applikationen skal sende data til din REST service provider.

# Opgave 8 Web-application, GET, statistik

Lav endnu en web-applikation med HTML og Typescript.

Din web-applikation skal bruge Vue.js og Bootstrap.

Web-applikationen skal hente data fra din REST service provider f.eks

* Data fra en bestemt by (*city*)
* Data fra en bestem størrelse husstand (*household*)

Bemærk at dette kan betyde at du skal udvide din REST service provider.

Din applikation skal have input felter til *city* og *household*.

Du skal vise de hentede CovidRecords som både *unordered list* og som *table*.