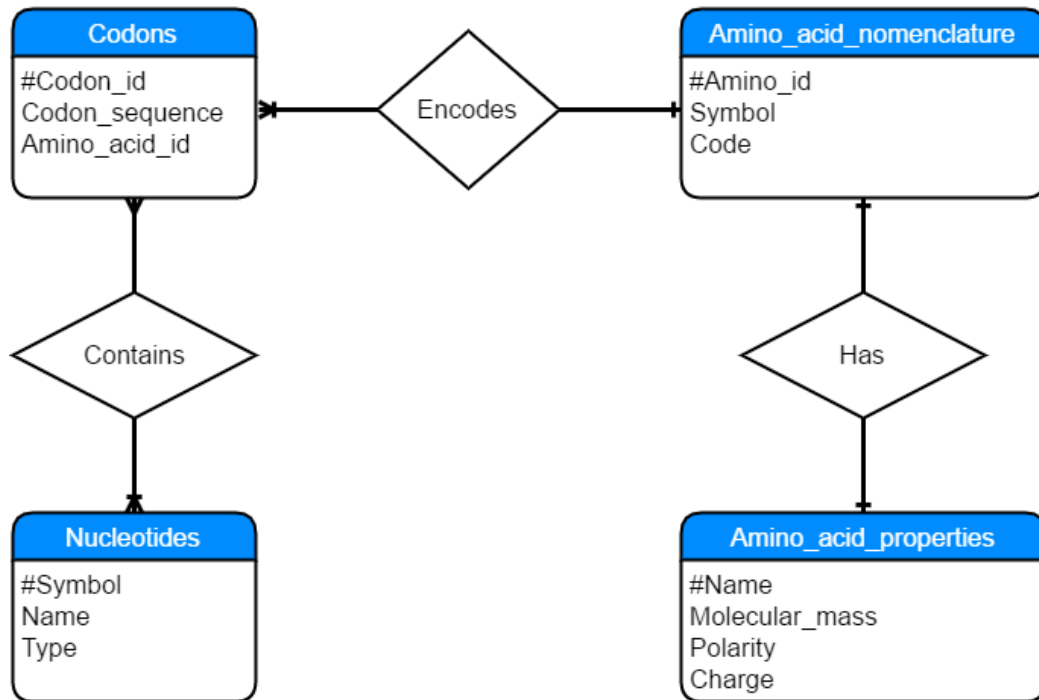


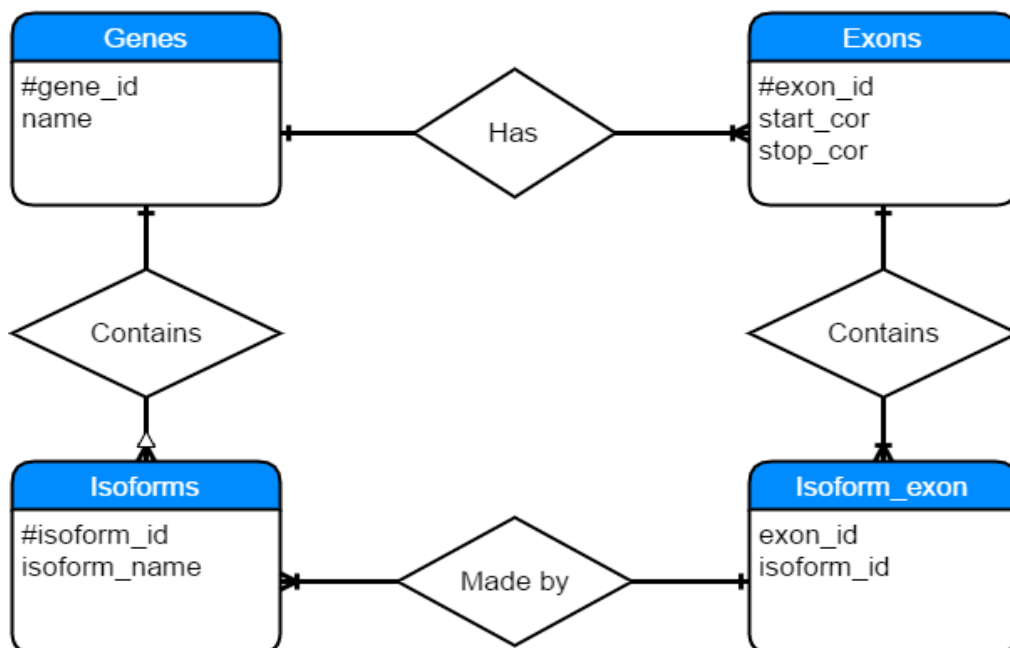
**INF115**  
**Oblig 2**  
**of002**

**Oppgave 1**



**Oppgave 2**

- i) Gene, Isoform, Exon, Isoform\_exon
- ii)

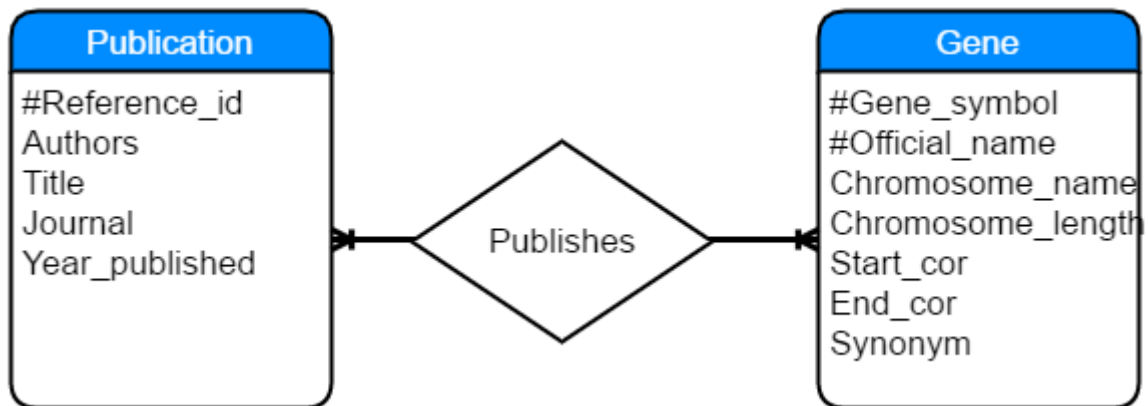


### Oppgave 3

i)

Publications, Gene

ii)



iii)

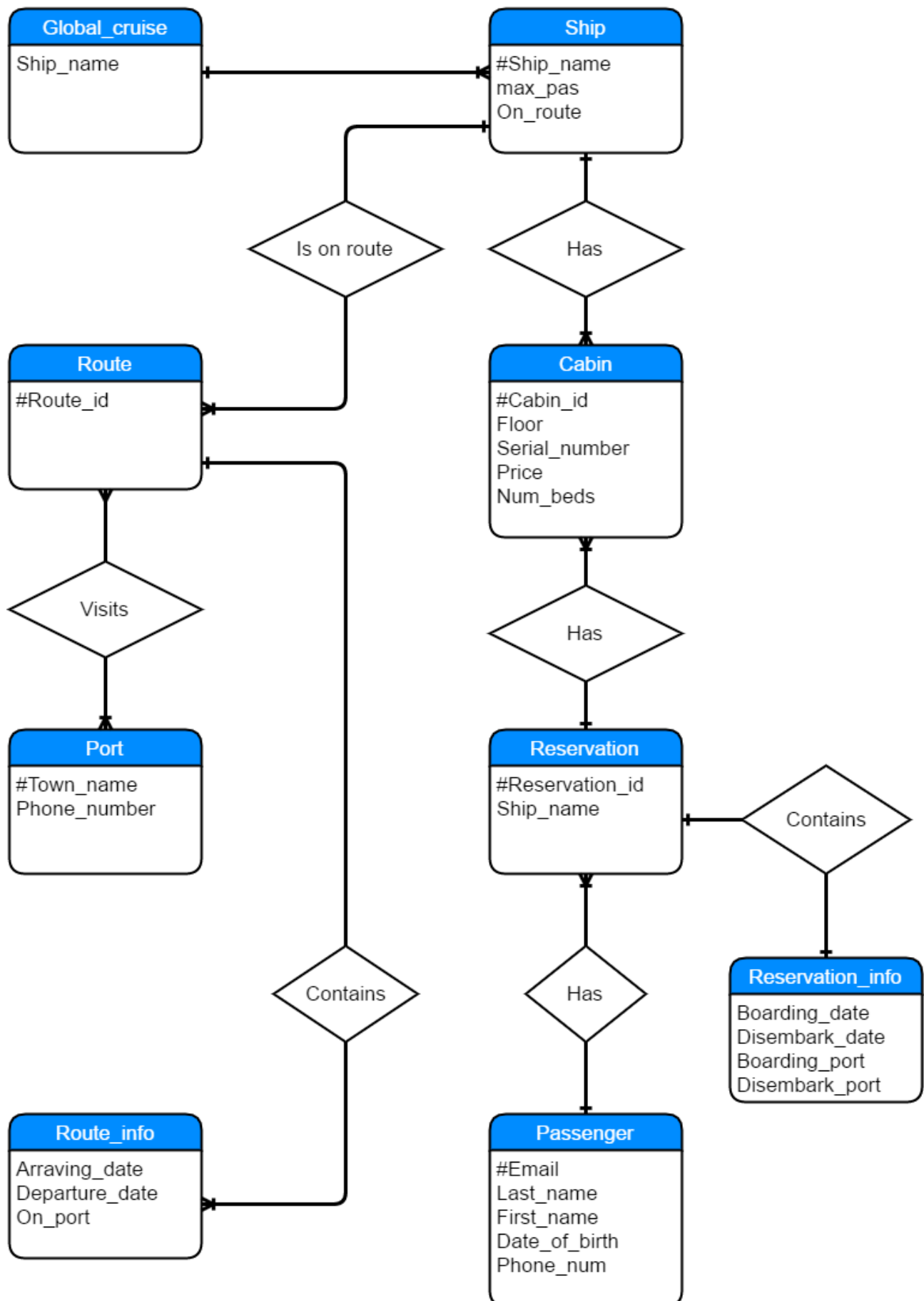
Gene\_symbol({Gene\_sym, Official\_name}, Synonym,  
Chromosome\_name, Chromosome\_length, Start\_cor, End\_cor)  
Reference({Reference\_id, Gene\_symbol}, Title, Author, Journal,  
Year\_published)

Med denne utføringen oppfyller tabellene førstegrads normalisering, men ikke andregrads. Dette fordi det fortsatt er partielle avhengigheter mellom tabellene.

iv)

Gene({Gene\_symbol, Official\_name}, Synonym\*,  
Chromosome\_name\*)  
Chromosome({Chromosome\_name, Chromosome\_length, Start\_cor,  
End\_cor})  
Synonyme({Synonyme, Gene\_symbol\*})  
Publications({Reference\_id, Authors\*, Title, Journal, Year\_published})  
Author({Reference\_id\*, Author})

## Oppgave 4



## Oppgave 5

i)

Problemet med denne løsningen er det at vi ikke vet hvilke biler som er ledig, og vi har heller ikke tilgang til å finne ut hvilken container som er på hvilken bil. Dette er også et problem fordi en bil blir stående som opptatt til gjennomføringen av assignmenten er ferdig, selv om den har vært ledig lenger. For å løse dette problemet kan vi utvide truck med Container Number. Dette gir oss muligheten til å vite om en bil er ledig eller ikke.

ii)

Ut i fra Registration\_number kan vi finne Registration\_year, Model, Maximun\_weight og Assignment\_number. Model bestemmer Maxumum weight.

iii)

Den minste supernøkkelen her er Registration\_number og derfor kandidatnøkkel.

iv)

Container type(#Type\_id, Type\_name, Max\_weight, Cubic\_quantity, Nightly\_rate)

Container(#Container\_number, Type\_id\*)

Customer(#Telephone\_number, Address)

Assignment(#Assignment\_number, Telephone\_number\*, Container\_number\*, Start\_date, End\_date)

Truck(#Registration\_number, Registration\_year, Model\*, Assignment\_number\*)

TruckModel(#Model, Max\_weight)