

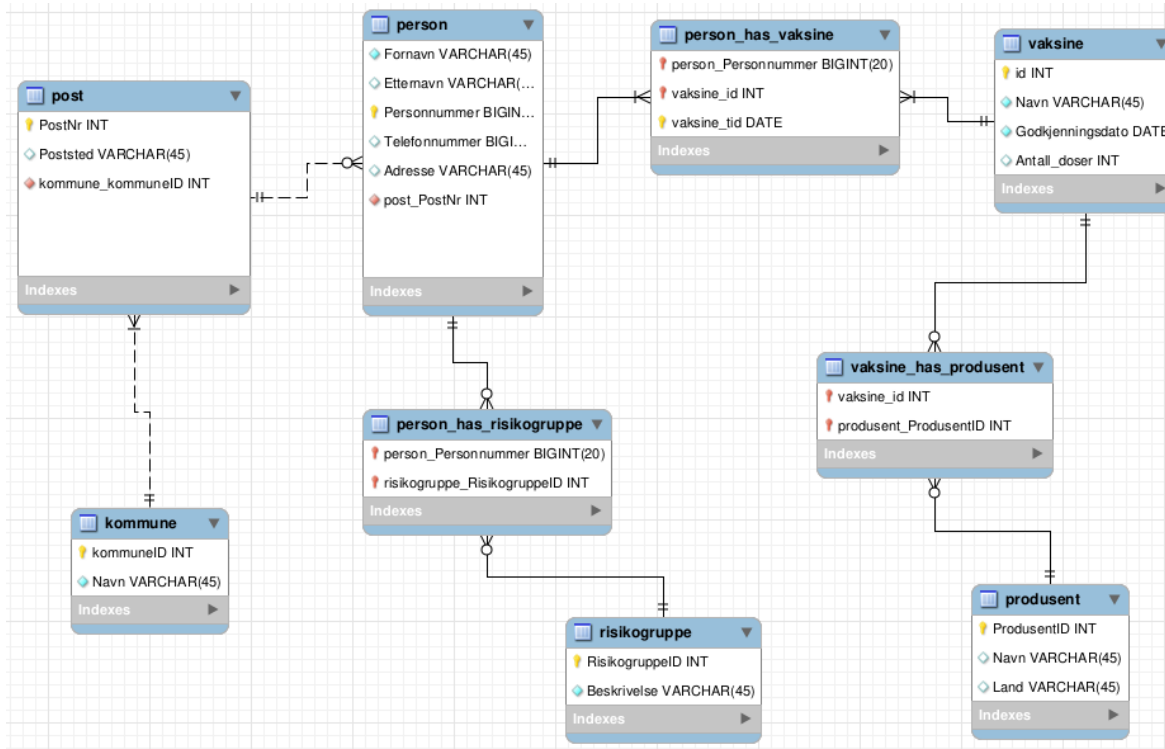
# Obilig 3\_2 - Fra datamodell til database

## - Yahye Abdi Ahmed

### Table of Contents

Model.....	2
Forward Engineering.....	2
Tabellene opprett fra modellen.....	7
Fyll av entitetene/tabellene med informasjon.....	8
Produsent entitet.....	8
Risikogruppe entitet.....	9
Kommune entitet.....	10
Post entitet.....	11
Vaksine entitet/tabell.....	12
person entitet.....	13
person_has_vaksine – person som har tatt korona vaksine.....	15
person_has_risikogruppe – risikogruppene personer er i.....	16

# Model



# Forward Engineering

SQL-script:

```
-- MySQL Workbench Forward Engineering
```

```
SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;  
SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;  
SET @OLD_SQL_MODE=@@SQL_MODE,  
SQL_MODE='ONLY_FULL_GROUP_BY, STRICT_TRANS_TABLES, NO_ZERO_IN_DATE, NO_ZERO_DATE, ERROR_FOR_DIVISION_BY_ZERO, NO_ENGINE_SUBSTITUTION';
```

```
-- Schema covid
```

```
-- Schema covid
```

```
CREATE SCHEMA IF NOT EXISTS `covid` DEFAULT CHARACTER SET utf8 ;  
USE `covid` ;
```

```
-- Table `covid`.`kommune`
```

```
CREATE TABLE IF NOT EXISTS `covid`.`kommune` (  
  `kommuneID` INT NOT NULL,  
  `Navn` VARCHAR(45) NOT NULL,  
  PRIMARY KEY (`kommuneID`))  
ENGINE = InnoDB  
AUTO_INCREMENT = 3001;
```

```
-- Table `covid`.`post`
```

```
CREATE TABLE IF NOT EXISTS `covid`.`post` (  
  `PostNr` INT NOT NULL,  
  `Poststed` VARCHAR(45) NULL,  
  `kommune_kommuneID` INT NOT NULL,  
  PRIMARY KEY (`PostNr`),  
  INDEX `fk_post_kommune1_idx` (`kommune_kommuneID` ASC) VISIBLE,  
  CONSTRAINT `fk_post_kommune1`  
    FOREIGN KEY (`kommune_kommuneID`)  
    REFERENCES `covid`.`kommune` (`kommuneID`)  
    ON DELETE NO ACTION
```

```
    ON UPDATE NO ACTION)
ENGINE = InnoDB;
```

```
-- Table `covid`.`person`
```

```
CREATE TABLE IF NOT EXISTS `covid`.`person` (
  `Fornavn` VARCHAR(45) NOT NULL,
  `Etternavn` VARCHAR(45) NULL,
  `Personnummer` BIGINT(20) NOT NULL,
  `Telefonnummer` BIGINT(20) NULL,
  `Adresse` VARCHAR(45) NULL,
  `post_PostNr` INT NOT NULL,
  PRIMARY KEY (`Personnummer`),
  INDEX `fk_person_post1_idx` (`post_PostNr` ASC) VISIBLE,
  CONSTRAINT `fk_person_post1`
    FOREIGN KEY (`post_PostNr`)
      REFERENCES `covid`.`post` (`PostNr`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8
COLLATE = utf8_bin
COMMENT = 'tabell for personer.';
```

```
-- Table `covid`.`vaksine`
```

```
CREATE TABLE IF NOT EXISTS `covid`.`vaksine` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `Navn` VARCHAR(45) NOT NULL,
  `Godkjenningsdato` DATE NOT NULL,
  `Antall_doser` INT NULL,
  PRIMARY KEY (`id`))
ENGINE = InnoDB;
```

```
-- Table `mydb`.`produsent`
```

```

CREATE TABLE IF NOT EXISTS `covid`.`produsent` (
  `ProdusentID` INT NOT NULL AUTO_INCREMENT,
  `Navn` VARCHAR(45) NULL,
  `Land` VARCHAR(45) NULL,
  PRIMARY KEY (`ProdusentID`))
ENGINE = InnoDB;

```

```

-----
-- Table `mydb`.`risikogruppe`
-----

```

```

CREATE TABLE IF NOT EXISTS `covid`.`risikogruppe` (
  `RisikogruppeID` INT NOT NULL AUTO_INCREMENT,
  `Beskrivelse` VARCHAR(45) NOT NULL,
  PRIMARY KEY (`RisikogruppeID`))
ENGINE = InnoDB;

```

```

-----
-- Table `mydb`.`vaksine_has_produsent`
-----

```

```

CREATE TABLE IF NOT EXISTS `covid`.`vaksine_has_produsent` (
  `vaksine_id` INT NOT NULL,
  `produsent_ProdusentID` INT NOT NULL,
  PRIMARY KEY (`vaksine_id`, `produsent_ProdusentID`),
  INDEX `fk_vaksine_has_produsent_produsent1_idx` (`produsent_ProdusentID` ASC) VISIBLE,
  INDEX `fk_vaksine_has_produsent_vaksine1_idx` (`vaksine_id` ASC) VISIBLE,
  CONSTRAINT `fk_vaksine_has_produsent_vaksine1`
    FOREIGN KEY (`vaksine_id`)
      REFERENCES `covid`.`vaksine` (`id`)
      ON DELETE NO ACTION
      ON UPDATE NO ACTION,
  CONSTRAINT `fk_vaksine_has_produsent_produsent1`
    FOREIGN KEY (`produsent_ProdusentID`)
      REFERENCES `covid`.`produsent` (`ProdusentID`)
      ON DELETE NO ACTION
      ON UPDATE NO ACTION)
ENGINE = InnoDB;

```

```

-----
-- Table `mydb`.`person_has_risikogruppe`
-----

```

```

-----
CREATE TABLE IF NOT EXISTS `covid`.`person_has_risikogruppe` (
  `person_Personnummer` BIGINT(20) NOT NULL,
  `risikogruppe_RisikogruppeID` INT NOT NULL,
  PRIMARY KEY (`person_Personnummer`, `risikogruppe_RisikogruppeID`),
  INDEX `fk_person_has_risikogruppe_risikogruppe1_idx` (`risikogruppe_RisikogruppeID` ASC) VISIBLE,
  INDEX `fk_person_has_risikogruppe_person1_idx` (`person_Personnummer` ASC) VISIBLE,
  CONSTRAINT `fk_person_has_risikogruppe_person1`
    FOREIGN KEY (`person_Personnummer`)
      REFERENCES `covid`.`person` (`Personnummer`)
      ON DELETE NO ACTION
      ON UPDATE NO ACTION,
  CONSTRAINT `fk_person_has_risikogruppe_risikogruppe1`
    FOREIGN KEY (`risikogruppe_RisikogruppeID`)
      REFERENCES `covid`.`risikogruppe` (`RisikogruppeID`)
      ON DELETE NO ACTION
      ON UPDATE NO ACTION)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8
COLLATE = utf8_bin;

```

```

-----
-- Table `mydb`.`person_has_vaksine`
-----

```

```

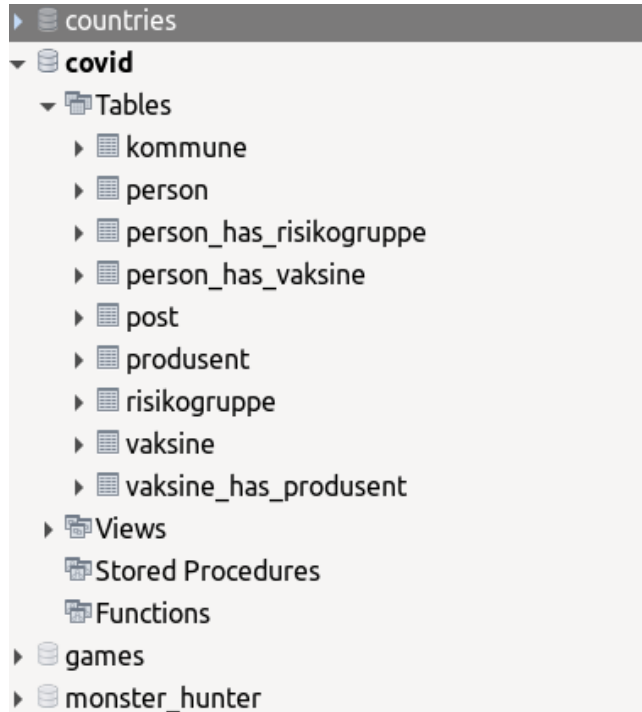
CREATE TABLE IF NOT EXISTS `covid`.`person_has_vaksine` (
  `person_Personnummer` BIGINT(20) NOT NULL,
  `vaksine_id` INT NOT NULL,
  `vaksine_tid` DATE NOT NULL,
  PRIMARY KEY (`person_Personnummer`, `vaksine_id`, `vaksine_tid`),
  INDEX `fk_person_has_vaksine_vaksine1_idx` (`vaksine_id` ASC) VISIBLE,
  INDEX `fk_person_has_vaksine_person1_idx` (`person_Personnummer` ASC) VISIBLE,
  CONSTRAINT `fk_person_has_vaksine_person1`
    FOREIGN KEY (`person_Personnummer`)
      REFERENCES `covid`.`person` (`Personnummer`)
      ON DELETE NO ACTION
      ON UPDATE NO ACTION,
  CONSTRAINT `fk_person_has_vaksine_vaksine1`
    FOREIGN KEY (`vaksine_id`)
      REFERENCES `covid`.`vaksine` (`id`)
      ON DELETE NO ACTION
      ON UPDATE NO ACTION)

```

```
ENGINE = InnoDB  
DEFAULT CHARACTER SET = utf8  
COLLATE = utf8_bin;
```

```
SET SQL_MODE=@OLD_SQL_MODE;  
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;  
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;
```

## Tabellene opprett fra modellen

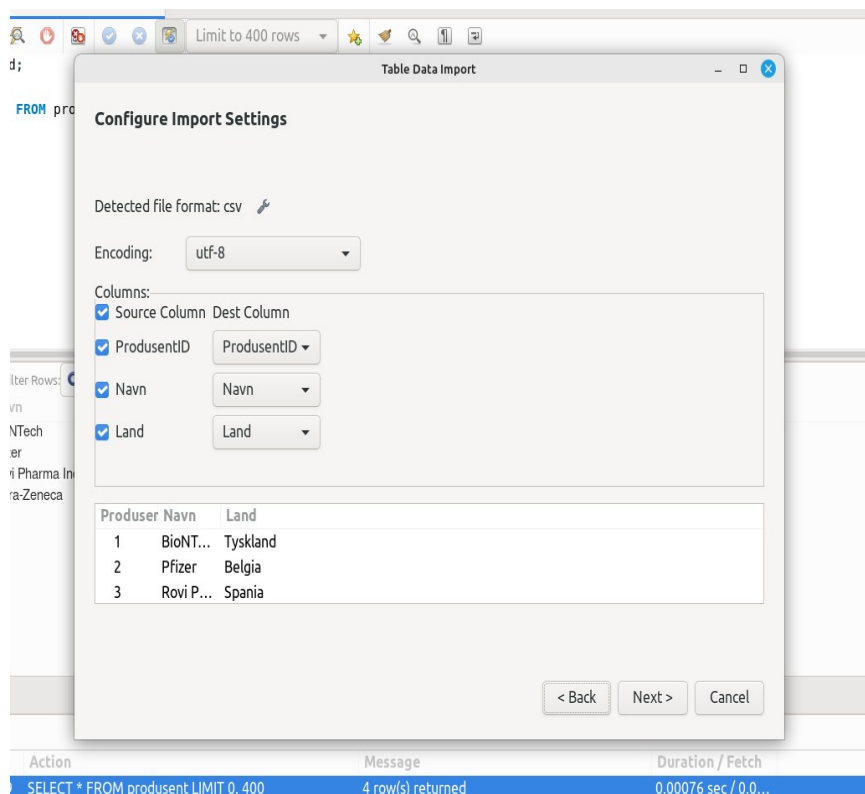


Vi kan se at **9** entitet/tabeller ble opprettet fra forward engineering

# Fyll av entitetene/tabellene med informasjon

## Produsent entitet

Med Table Date Import Wizard.



SQL-query:

**SELECT \* FROM** produsent;

Result Grid			
#	ProdusentID	Navn	Land
1	1	BioNTech	Tyskland
2	2	Pfizer	Belgia
3	3	Rovi Pharma Industrial Services	Spania
4	4	Astra-Zeneca	Storbritannia
*	NULL	NULL	NULL



## Risikogruppe entitet

Table Data Import

Limit to 400 rows

### Configure Import Settings

Detected file format: csv

Encoding: utf-8

Columns:

<input checked="" type="checkbox"/> Source Column	Dest Column
<input checked="" type="checkbox"/> RisikogruppeID	RisikogruppeID
<input checked="" type="checkbox"/> Beskrivelse	Beskrivelse

Risikogr Beskrivelse

1	Sykehjem
2	Helsepersonell
3	Risikogruppe

< Back   Next >   Cancel

SQL-spørring:

**SELECT** \* **FROM** risikogruppe;

Result Grid

Filter Rows:

Edit:

#	RisikogruppeID	Beskrivelse
1	1	Sykehjem
2	2	Helsepersonell
3	3	Risikogruppe
4	4	85+
5	5	75-84
6	6	65-74
7	7	55-64
8	8	45-54
9	9	18-44
*	NULL	NULL

# Kommune entitet

Table Data Import

Configure Import Settings

Detected file format: csv

Encoding: utf-8

Columns:

Source Column	Dest Column
<input checked="" type="checkbox"/> KommuneID	KommuneID
<input checked="" type="checkbox"/> Navn	Navn

Kommune Navn

3001	Halden
3002	Moss
3003	Sarpsborg

< Back Next > Cancel

SQL-spørring:

Attributtet "KommuneID" måtte endres fordi den begynte med liten bokstav i ER-modellen.

#change column name in kommune table

```
Alter table kommune rename column kommuneID TO KommuneID;
```

```
SELECT * FROM kommune;
```

Result Grid

Filter Rows:

#	KommuneID	Navn
1	3001	Halden
2	3002	Moss
3	3003	Sarpsborg
4	3004	Fredrikstad
5	3011	Hvaler
6	3012	Aremark
7	3013	Marker
8	3014	Indre Østfold
9	3015	Skiptvedt
10	3016	Rakkestad
11	3017	Råde
12	3018	Våler

# Post entitet

Table Data Import

Configure Import Settings

Detected file format: csv

Encoding: utf-8

Columns:

Source Column	Dest Column
<input checked="" type="checkbox"/> PostNr	PostNr
<input checked="" type="checkbox"/> Poststed	Poststed
<input checked="" type="checkbox"/> Kommune_KommuneID	Kommune_KommuneID

PostNr	Poststed	Kommune_KommuneID
1501	MOSS	3002
1502	MOSS	3002
1503	MOSS	3002

< Back   Next >   Cancel

+

SQL-spørring:

#change coloumn to match the one from the csv

**Alter table** post **rename column** kommune\_kommuneID  
**TO** Kommune\_KommuneID;

#post table

**SELECT** \* **from** post;

Result Grid

Filter Rows:

#	id	PostNr	Poststed	Kommune_KommuneID
1	237	1501	MOSS	3002
2	238	1502	MOSS	3002
3	239	1503	MOSS	3002
4	240	1504	MOSS	3002
5	241	1506	MOSS	3002
6	242	1508	MOSS	3002
7	243	1509	MOSS	3002
8	244	1510	MOSS	3002
9	245	1511	MOSS	3002
10	246	1512	MOSS	3002
11	247	1513	MOSS	3002
12	248	1514	MOSS	3002

post 8

## Vaksine entitet/tabell

SQL-spørring for endring insert av data i vaksine tabell:

```
#insert values in vaksine
INSERT INTO vaksine(Navn, Godkjenningsdato, Antall_dose)
VALUES
("pfizer", "2020, 11, 20", 4)
;

INSERT INTO vaksine(Navn, Godkjenningsdato, Antall_dose)
VALUES
("Oxford-AstraZeneca", "2021-03-26", 4),
("Moderna", "2021-04-12", 3)
;

#connect vaksine to correct producer using middle table
#here for example pfizer has 2 producers.
INSERT INTO vaksine_has_producent(vaksine_id, produsent_id)
VALUES
(1, 1),
(1, 2),
(2, 4),
(3, 3)
;

SELECT * FROM produsent;

SELECT * FROM vaksine;

select * from vaksine_has_producent;
```

## Vaksine tabell

Result Grid

Filter Rows:

Edit:

#	id	Navn	Godkjenningsdat	Antall_dose
1	1	Comernaty	2020-11-20	4
2	2	Oxford-AstraZeneca	2021-03-26	4
3	3	Moderna	2021-04-12	3
*	NULL	NULL	NULL	NULL

## vaksine\_has\_produsent

Result Grid

Filter Rows:

Edit:

#	id	vaksine_id	produsent_id
1	1	1	1
2	2	1	2
3	3	2	4
4	4	3	3
*	NULL	NULL	NULL

## produsent

Result Grid

Filter Rows:

Edit:

#	ProdusentID	Navn	Land
1	1	BioNTech	Tyskland
2	2	Pfizer	Belgia
3	3	Rovi Pharma Industrial Services	Spania
4	4	Astra-Zeneca	Storbritannia
*	NULL	NULL	NULL

## person entitet

I person entitet skal vi lage minimum 10 rader.

**SQL-spørring som setter inn 10 rader i person entitet:**

```
INSERT INTO person(Fornavn, Etternavn, personnummer, Telefonnummer, Adresse, post_postNr)
VALUES
("kabada", "kii", 21099238795, 4799003330, "fake street", 1602),
("lala", "lire", 23073226317, 4748332264, "prog2", 1624),
("mr", "anderson", 21029633913, 4759991463, "grass", 1641),
("omba", "lumba", 29109318297, 4741088114, "mountains", 1657),
("nisar", "read", 01019416395, 4791735957, "dondurma", 1683),
("billie", "bin", 21114739550, 4740114532, "stalingrad", 1701),
("tic", "tac", 26034941296, 4740481251, "keke", 1742),
("mara", "kilini", 30077102244, 4748113068, "gate kjar", 1743),
("bam", "barabum", 04127922824, 4747470572, "tnt land", 1751),
("mikel", "juqson", 18108218904, 4746327461, "smooth", 1757)
;
```

```
SELECT *  
FROM person;
```

Result Grid							
Filter Rows:				Edit:		Export/Imp	
#	Fornavn	Etternavn	Personnumme	Telefonnumme	Adresse	post_PostN	
1	nisar	read	1019416395	4791735957	dondurma	1683	
2	bam	barabum	4127922824	4747470572	tnt land	1751	
3	mikel	juqson	18108218904	4746327461	smooth	1757	
4	mr	anderson	21029633913	4759991463	grass	1641	
5	kabada	kii	21099238795	4799003330	fake street	1602	
6	billie	bin	21114739550	4740114532	stalingrad	1701	
7	lala	lire	23073226317	4748332264	prog2	1624	
8	tic	tac	26034941296	4740481251	keke	1742	
9	omba	lumba	29109318297	4741088114	mountains	1657	
10	mara	kilini	30077102244	4748113068	gate kjar	1743	
*	NULL	NULL	NULL	NULL	NULL	NULL	

# person\_has\_vaksine – person som har tatt korona vaksine

## SQL-spørring som setter inn entitet person\_has\_vaksine data

Person blir koblet til vaksine tabellen via mellom tabellen

```
#people who have taken the vax and they took it
# 1: pfizer, 2: astra, 3: moderna
INSERT INTO person_has_vaksine(person_Personnummer,vaksine_id,vaksine_tid)
VALUES
(1019416395,3,"2021-05-28"),
(4127922824,1,"2020-08-22"),
(4127922824,1,"2021-01-12"),
(4127922824,1,"2022-10-10"),
(18108218904,2,"2020-11-25"),
(21029633913,3,"2021-03-14"),
(21029633913,3,"2021-12-28"),
(21099238795,2,"2022-01-21"),
(21114739550,1,"2020-11-18"),
(23073226317,3,"2021-01-23"),
(23073226317,1,"2021-11-05"),
(23073226317,3,"2022-06-14"),
(26034941296,2,"2022-03-15"),
(30077102244,1,"2020-05-11"),
(30077102244,1,"2021-01-19")
;

SELECT *
FROM person_has_vaksine;
```

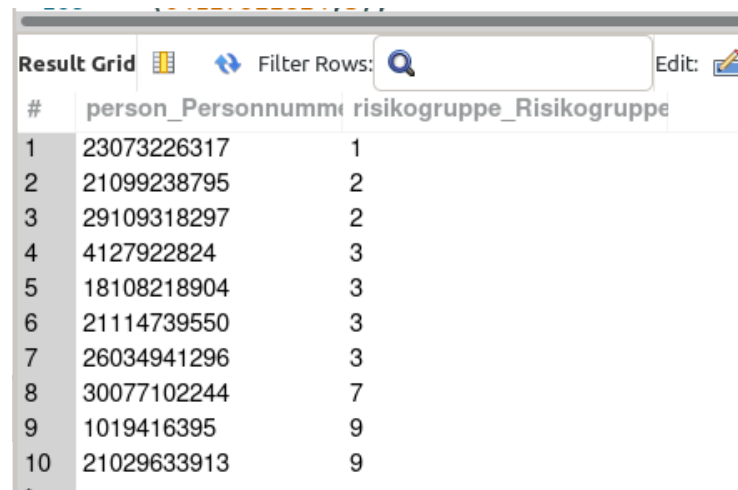
#	person_Personnummer	vaksine_id	vaksine_tid
1	4127922824	1	2020-08-22
2	4127922824	1	2021-01-12
3	4127922824	1	2022-10-10
4	21114739550	1	2020-11-18
5	23073226317	1	2021-11-05
6	30077102244	1	2020-05-11
7	30077102244	1	2021-01-19
8	18108218904	2	2020-11-25
9	21099238795	2	2022-01-21
10	26034941296	2	2022-03-15
11	1019416395	3	2021-05-28
12	21029633913	3	2021-03-14
13	21029633913	3	2021-12-28
14	23073226317	3	2021-01-23
15	23073226317	3	2022-06-14
*	NULL	NULL	NULL

## person\_has\_risikogruppe – risikogruppene personer er i

### SQL-spørring som setter person i riktig risikogruppe

```
#insert data into person_has_risikogruppe
INSERT INTO person_has_risikogruppe(person_Personnummer, risikogruppe_RisikogruppeID)
VALUES
(21099238795, 2),
(23073226317, 1),
(21029633913, 9),
(29109318297, 2),
(01019416395, 9),
(21114739550, 3),
(26034941296, 3),
(30077102244, 7),
(04127922824, 3),
(18108218904, 3)
;
```

```
SELECT *
FROM person_has_risikogruppe;
```



The screenshot shows a database query result grid with the following data:

#	person_Personnummer	risikogruppe_Risikogruppe
1	23073226317	1
2	21099238795	2
3	29109318297	2
4	4127922824	3
5	18108218904	3
6	21114739550	3
7	26034941296	3
8	30077102244	7
9	1019416395	9
10	21029633913	9