Obligatorisk innlevering Noora Rehim Lindeflaten

OPPGÅVE 1

```
# 1a
SELECT COUNT(Codon_id) as codons FROM Codons;

#1b
SELECT * FROM Amino_acid_properties WHERE charge = "Positive" AND
Molecular_mass > 150;

#1c
SELECT * FROM Nucleotides WHERE Type = "Purine" ORDER BY Symbol;

#1d
SELECT Codon_sequence FROM Codons WHERE Position2 = Position3;

#1e
SELECT Codon_sequence, Amino_acid_id FROM Codons GROUP BY
Amino_acid_id HAVING COUNT(*) = 1;
```

OPPGÅVE 2

```
#2a
CREATE TABLE Amino_acid_nomenclature
(
   Amino_id VARCHAR(5) UNIQUE,
   Symbol CHAR(1) NULL,
   Name VARCHAR(40) NULL,
   Code VARCHAR(4), CONSTRAINT Amino_id_pk PRIMARY KEY(Amino_id),
   CONSTRAINT Name_fk FOREIGN KEY(Name) REFERENCES
Amino_acid_properties (Name));

#2b
INSERT INTO amino_acid_nomenclature VALUES("a1", "A", "Alanine", "Ala");
INSERT INTO amino_acid_nomenclature VALUES("a2", "C", "Cysteine",
"Cys");
INSERT INTO amino_acid_nomenclature VALUES("a3", "D", "Aspartic acid",
"Asp");
INSERT INTO amino_acid_nomenclature VALUES("a4", "E", "Glutamic acid",
"Glu");
INSERT INTO amino_acid_nomenclature VALUES("a5", "F", "Phenylalanine",
"Phe");
INSERT INTO amino_acid_nomenclature VALUES("a6", "G", "Glycine",
"Gly");
INSERT INTO amino_acid_nomenclature VALUES("a7", "H", "Histidine",
"His");
INSERT INTO amino_acid_nomenclature VALUES("a8", "I", "Isoleucine",
"Ile");
```

```
INSERT INTO amino_acid_nomenclature VALUES("a9", "K", "Lysine", "Lys");
INSERT INTO amino_acid_nomenclature VALUES("a10", "L", "Leucine",
"Leu");
INSERT INTO amino_acid_nomenclature VALUES("a11", "M", "Methionine",
"Met");
INSERT INTO amino_acid_nomenclature VALUES("a12", "N", "Asparagine",
"Asn");
INSERT INTO amino_acid_nomenclature VALUES("a13", "P", "Proline",
"Pro");
INSERT INTO amino_acid_nomenclature VALUES("a14", "Q", "Glutamine",
"Gln");
INSERT INTO amino_acid_nomenclature VALUES("a15", "R", "Arginine",
"Arg");
INSERT INTO amino_acid_nomenclature VALUES("a16", "S", "Serine",
"Ser");
INSERT INTO amino_acid_nomenclature VALUES("a17", "T", "Threonine",
"Thr");
INSERT INTO amino_acid_nomenclature VALUES("a18", "V", "Valine", "Val");
INSERT INTO amino_acid_nomenclature VALUES("a19", "W", "Tryptophan",
"Trp");
INSERT INTO amino_acid_nomenclature VALUES("a20", "Y", "Tyrosine",
"Tyr");
INSERT INTO amino_acid_nomenclature VALUES("a21", NULL, NULL,
"Stop");
INSERT INTO amino_acid_nomenclature VALUES("a22", NULL, NULL,
"Stop");
INSERT INTO amino_acid_nomenclature VALUES("a23", NULL, NULL,
"Stop");

#2c
2c i)
ALTER TABLE amino_acid_properties
ADD CONSTRAINT number_check CHECK (Molecular_mass > 70 AND
Molecular_mass < 210);
   ii)
ALTER TABLE amino_acid_properties
ADD CONSTRAINT charge_check CHECK (Charge = "Positive" OR Charge =
"Negative" OR Charge = "uncharged");

#2d
ALTER TABLE Codons
ADD CONSTRAINT Amino_acid_pk
FOREIGN KEY (Amino_acid_id) REFERENCES
```

```
Amino_acid_nimenclature(Amino_id);

#3a

SELECT c.*
FROM codons c
JOIN amino_acid_nomenclature a1 ON a1.Amino_id = c.Amino_acid_id
WHERE a1.Code = "Stop";

#3b

SELECT c.Codon_sequence
FROM codons c
JOIN nucleotides n ON n.Symbol = c.Position1
WHERE n.Name = "Cytosine"

#3c

SELECT c.Codon_sequence, a2.Molecular_mass
FROM codons c
JOIN amino_acid_nomenclature a1 ON a1.Amino_id = c.Amino_acid_id
JOIN amino_acid_properties a2 ON a2.Name = a1.Name
ORDER BY a2.Molecular_mass ASC;

#3d

SELECT COUNT(a2.Charge) AS Count_Amino_Uncharged
FROM codons c
JOIN amino_acid_nomenclature a1 ON a1.Amino_id = c.Amino_acid_id
JOIN amino_acid_properties a2 ON a2.Name = a1.Name
WHERE c.Position3 = "A" AND a2.Charge = "uncharged";

#3e


SELECT c.Codon_sequence, a1.Name
FROM codons c
JOIN amino_acid_nomenclature a1 ON a1.Amino_id = c.Amino_acid_id
JOIN amino_acid_properties a2 ON a2.Name = a1.Name
WHERE a2.Charge = "uncharged" AND (a2.Molecular_mass > 130 AND
a2.Molecular_mass < 150);

---------------------------------
```

```
#4a

SELECT
   SUM(CASE WHEN Type = "purine" THEN 1 ELSE 0 END) AS
Purine_Count,
   SUM(case when Type = "pyrimidine" THEN 1 ELSE 0 END) AS
Purimidine_Count
FROM nucleotides;

#4b

SELECT a1.Symbol
FROM codons c
JOIN amino_acid_nomenclature a1 ON a1.Amino_id = c.Amino_acid_id
WHERE c.Position1 = c.Position2 AND c.Position1 = c.Position3
ORDER BY a1.Name;

#4c


SELECT c.Codon_sequence
FROM codons c
JOIN amino_acid_nomenclature a1 ON a1.Amino_id = c.Amino_acid_id
JOIN amino_acid_properties a2 ON a2.Name = a1.Name
JOIN nucleotides n ON n.Symbol = c.Position1
WHERE a2.Polarity = "polar" AND a2.Name LIKE "%ine" AND n.Type =
"purine";

#4d

SELECT
   SUM(CASE WHEN a2.Polarity = "polar" THEN 1 ELSE 0 END) AS Polar,
   SUM(CASE WHEN a2.Polarity = "non polar" THEN 1 ELSE 0 END) AS
Nonpolar
FROM codons c
JOIN amino_acid_nomenclature a1 ON a1.Amino_id = c.Amino_acid_id
JOIN amino_acid_properties a2 ON a2.Name = a1.Name;


#4e

SELECT
```

```
   COUNT(CASE
        WHEN a2.Polarity = "polar" AND a2.Charge = "uncharged" THEN 1
END) AS "Polar/Uncharged",
   COUNT(CASE
        WHEN a2.Polarity = "polar" AND a2.Charge = "positive" THEN 1 END)
AS "Polar/Positive",
   COUNT(CASE
        WHEN a2.Polarity = "polar" AND a2.Charge = "negative" THEN 1 END)
AS "Polar/Negative",
   COUNT(CASE
        WHEN a2.Polarity = "non polar" AND a2.Charge = "uncharged" THEN
1 END) AS "Nonpolar/Uncharged"
FROM codons c
JOIN amino_acid_nomenclature a1 ON a1.Amino_id = c.Amino_acid_id
JOIN amino_acid_properties a2 ON a2.Name = a1.Name;
```