Task1

```sql
SELECT name, price
FROM (
  SELECT *
  FROM Products
  WHERE
    price > 100 AND
    name like "L%"
) as subquery;
```

Auto Commit | Manual

	name	price
1	Laptop	999.99

1 | 0 affected | 5ms

```sql
SELECT name, price
FROM Products
WHERE
  price > 100 AND
  name like "L%";
```

Auto Commit | Manual

	name	price
1	Laptop	999.99

1 | 0 affected | 4ms

Fig.1 — nested query.
Fig.2 — optimized query.

Executed on: Beekeper Studio (database client)
Time delta: 1ms.

Conclusion: Nested queries cause performance bottlenecks which can be even more pronounced on large datasets.

Task 2

Query #1

```sql
SELECT c.name, o.order_date
FROM
  Customers as c
INNER JOIN
  Orders as o
  ON o.customer_id = c.customer_id;
```

Auto Commit | Manual

	name	order_date
1	Alice Chen	2024-03-15
2	Alice Chen	2024-03-17
3	Bob Smith	2024-03-16
4	Bob Smith	2024-03-19
5	Carol Davis	2024-03-17
6	David Wilson	2024-03-18
7	Eva Brown	2024-03-18

7 | 0 affected | 6ms

Task 3

```sql
SELECT *
FROM
  Orders, Products;
```

Auto Commit | Manual

	order_id	customer_id	order_date	product_id	name	price
1	1	1	2024-03-15	5	USB-C Hub	89.99
2	1	1	2024-03-15	4	Monitor	300.00
3	1	1	2024-03-15	3	Keyboard	75.00
4	1	1	2024-03-15	2	Mouse	25.50
5	1	1	2024-03-15	1	Laptop	999.99
6	2	2	2024-03-16	5	USB-C Hub	89.99
7	2	2	2024-03-16	4	Monitor	300.00
8	2	2	2024-03-16	3	Keyboard	75.00
9	2	2	2024-03-16	2	Mouse	25.50
10	2	2	2024-03-16	1	Laptop	999.99
11	3	1	2024-03-17	5	USB-C Hub	89.99
12	3	1	2024-03-17	4	Monitor	300.00
13	3	1	2024-03-17	3	Keyboard	75.00
14	3	1	2024-03-17	2	Mouse	25.50
15	3	1	2024-03-17	1	Laptop	999.99

35 | 0 affected | 5ms

Task 4

```sql
CREATE TABLE Department (
   dept_id     INT,
   dept_name   VARCHAR(50) NOT NULL,
   building    VARCHAR(50),
   budget      DECIMAL(12, 2),

   PRIMARY KEY (dept_id)
);

CREATE TABLE Instructor (
   instructor_id   INT,
   name            VARCHAR(50) NOT NULL,
   dept_id         INT NULL,
   salary          DECIMAL(12, 2),
   hire_date       DATE DEFAULT (CURRENT_DATE()),

   PRIMARY KEY (instructor_id),

   FOREIGN KEY (dept_id)
       REFERENCES Department(dept_id)
       ON DELETE SET NULL
);

CREATE TABLE Course (
   course_id       VARCHAR(50),
   title           VARCHAR(50) NOT NULL, -- does not make sense if title is missing
   dept_id         INT,
   credits         INT CHECK (credits >= 0),
   instructor_id   INT NULL,

   PRIMARY KEY (course_id),

   FOREIGN KEY (dept_id)
       REFERENCES Department(dept_id)
       ON DELETE CASCADE,

   FOREIGN KEY (instructor_id)
       REFERENCES Instructor(instructor_id)
       ON DELETE SET NULL
);
```

```sql
CREATE TABLE Student (
   student_id      INT,
   name            VARCHAR(50) NOT NULL,
   dept_id         INT NULL,
   enrollment_year INT,
   total_credits   INT CHECK (total_credits >= 0),
   advisor_id      INT NULL,

   PRIMARY KEY (student_id),

   FOREIGN KEY (dept_id)
       REFERENCES Department(dept_id)
       ON DELETE SET NULL,
   FOREIGN KEY (advisor_id)
       REFERENCES Instructor(instructor_id)
       ON DELETE SET NULL
);
CREATE TABLE Takes (
   student_id      INT,
   course_id       VARCHAR(50),
   semester        VARCHAR(50),
   year            INT,
   grade           CHAR(2),

   PRIMARY KEY (student_id, course_id, semester, year),

   FOREIGN KEY (student_id)
       REFERENCES Student(student_id)
       ON DELETE CASCADE,
   FOREIGN KEY (course_id)
       REFERENCES Course(course_id)
       ON DELETE RESTRICT
);

CREATE TABLE Prerequisite (
   course_id       VARCHAR(50),
   prereq_id       VARCHAR(50),
   min_grade       CHAR(2),

   PRIMARY KEY (course_id, prereq_id),
   FOREIGN KEY (course_id)
       REFERENCES Course(course_id)
       ON DELETE CASCADE,
   FOREIGN KEY (prereq_id)
       REFERENCES Course(course_id)
       ON DELETE RESTRICT
);
```

Task 5

1.

```sql
SELECT
  Student.*,
  Advisor.name, Advisor.dept_id,
  Advisor.salary, Advisor.hire_date
FROM
  Student
INNER JOIN
  Instructor as Advisor
  ON Student.advisor_id = Advisor.instructor_id;
```

Auto Commit | Manual

	student_id	name	dept_id	enrollment_year	total_credits	advisor_id	name	dept_id	salary	hire_date
1	1	John Anderson	1	2022	30	1	Dr. Alice Smith	1	85000.00	2018-05-15
2	2	Sarah Parker	1	2022	28	1	Dr. Alice Smith	1	85000.00	2018-05-15
3	8	Sophia Wilson	1	2021	90	2	Dr. Bob Johnson	1	92000.00	2015-08-20
4	3	Michael Chen	2	2021	75	3	Dr. Carol Williams	2	78000.00	2020-01-10
5	4	Jessica Taylor	3	2023	15	4	Dr. David Brown	3	81000.00	2019-03-22
6	6	Olivia Martinez	4	2023	12	5	Dr. Emma Davis	4	76000.00	2021-11-05
7	7	Daniel Kim	5	2022	45	7	Dr. Frank Wilson	5	72000.00	2022-02-14

2.

```sql
SELECT
  Student.*,
  Advisor.name, Advisor.dept_id,
  Advisor.salary, Advisor.hire_date
FROM
  Student
LEFT JOIN
  Instructor as Advisor
  ON Student.advisor_id = Advisor.instructor_id;
```

Auto Commit | Manual

	student_id	name	dept_id	enrollment_year	total_credits	advisor_id	name	dept_id	salary	hire_date
1	1	John Anderson	1	2022	30	1	Dr. Alice Smith	1	85000.00	2018-05-15
2	2	Sarah Parker	1	2022	28	1	Dr. Alice Smith	1	85000.00	2018-05-15
3	3	Michael Chen	2	2021	75	3	Dr. Carol Williams	2	78000.00	2020-01-10
4	4	Jessica Taylor	3	2023	15	4	Dr. David Brown	3	81000.00	2019-03-22
5	5	Kevin Rodrig…	(NULL)	2022	20	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
6	6	Olivia Martinez	4	2023	12	5	Dr. Emma Davis	4	76000.00	2021-11-05
7	7	Daniel Kim	5	2022	45	7	Dr. Frank Wilson	5	72000.00	2022-02-14
8	8	Sophia Wilson	1	2021	90	2	Dr. Bob Johnson	1	92000.00	2015-08-20

3.

```sql
SELECT
  Advisor.*,
  Student.student_id,
  Student.name, Student.dept_id,
  Student.enrollment_year, Student.total_credits
FROM
  Student
RIGHT JOIN
  Instructor as Advisor
  ON Student.advisor_id = Advisor.instructor_id;

```

Auto Commit | Manual

	instructor_id	name	dept_id	salary	hire_date	student_id	name	dept_id	enrollment_year	total_credits
1	1	Dr. Alice Smith	1	85000.00	2018-05-15	1	John Anderson	1	2022	30
2	1	Dr. Alice Smith	1	85000.00	2018-05-15	2	Sarah Parker	1	2022	28
3	2	Dr. Bob Johnson	1	92000.00	2015-08-20	8	Sophia Wilson	1	2021	90
4	3	Dr. Carol Williams	2	78000.00	2020-01-10	3	Michael Chen	2	2021	75
5	4	Dr. David Brown	3	81000.00	2019-03-22	4	Jessica Taylor	3	2023	15
6	5	Dr. Emma Davis	4	76000.00	2021-11-05	6	Olivia Martinez	4	2023	12
7	7	Dr. Frank Wilson	5	72000.00	2022-02-14	7	Daniel Kim	5	2022	45
8	8	Dr. Grace Lee	(NULL)	68000.00	2023-08-01	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)

4.

```sql
SELECT
  *
FROM
  Student
LEFT JOIN
  Instructor as Advisor
  ON Student.advisor_id = Advisor.instructor_id

UNION

SELECT
  *
FROM
  Student
RIGHT JOIN
  Instructor as Advisor
  ON Student.advisor_id = Advisor.instructor_id;

```

Auto Commit | Manual

	student_id	name	dept_id	enrollment_year	total_credits	advisor_id	instructor_id	name	dept_id	salary	hire_date
1	1	John Anderson	1	2022	30	1	1	Dr. Alice Smith	1	85000.00	2018-05-15
2	2	Sarah Parker	1	2022	28	1	1	Dr. Alice Smith	1	85000.00	2018-05-15
3	3	Michael Chen	2	2021	75	3	3	Dr. Carol Williams	2	78000.00	2020-01-10
4	4	Jessica Taylor	3	2023	15	4	4	Dr. David Brown	3	81000.00	2019-03-22
5	5	Kevin Rodrig…	(NULL)	2022	20	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
6	6	Olivia Martinez	4	2023	12	5	5	Dr. Emma Davis	4	76000.00	2021-11-05
7	7	Daniel Kim	5	2022	45	7	7	Dr. Frank Wilson	5	72000.00	2022-02-14
8	8	Sophia Wilson	1	2021	90	2	2	Dr. Bob Johnson	1	92000.00	2015-08-20
9	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	8	Dr. Grace Lee	(NULL)	68000.00	2023-08-01

5.

```sql
SELECT
  Student.*,
  Department.*,
  Course.*
FROM
  Student
LEFT JOIN
  Department
  ON Student.dept_id = Department.dept_id
LEFT JOIN
  Takes
  ON Takes.student_id = Student.student_id
LEFT JOIN
  Course
  ON Course.course_id = Takes.course_id;


```

Auto Commit | Manual

	student_id	name	dept_id	enrollment_year	total_credits	advisor_id	dept_id	dept_name	building	budget	course_id	title	dept_id	credits	instructor_id
1	1	John Anderson	1	2022	30	1	1	Computer Science	Engineering	500000.00	CS101	Intro…	1	4	1
2	1	John Anderson	1	2022	30	1	1	Computer Science	Engineering	500000.00	CS201	Data…	1	3	2
3	1	John Anderson	1	2022	30	1	1	Computer Science	Engineering	500000.00	MATH101	Calc…	2	4	3
4	2	Sarah Parker	1	2022	28	1	1	Computer Science	Engineering	500000.00	CS101	Intro…	1	4	1
5	2	Sarah Parker	1	2022	28	1	1	Computer Science	Engineering	500000.00	CS301	Algo…	1	3	2
6	3	Michael Chen	2	2021	75	3	2	Mathematics	Science	300000.00	MATH101	Calc…	2	4	3
7	3	Michael Chen	2	2021	75	3	2	Mathematics	Science	300000.00	MATH201	Line…	2	3	(NULL)
8	4	Jessica Taylor	3	2023	15	4	3	Physics	Science	350000.00	PHYS101	Mec…	3	4	4
9	5	Kevin Rodriguez	(NULL)	2022	20	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
10	6	Olivia Martinez	4	2023	12	5	4	Biology	Life Sciences	400000.00	BIOL101	Biol…	4	4	5
11	7	Daniel Kim	5	2022	45	7	5	Chemistry	Science	280000.00	CHEM101	Gen…	5	4	7
12	8	Sophia Wilson	1	2021	90	2	1	Computer Science	Engineering	500000.00	CS101	Intro…	1	4	1
13	8	Sophia Wilson	1	2021	90	2	1	Computer Science	Engineering	500000.00	CS201	Data…	1	3	2
14	8	Sophia Wilson	1	2021	90	2	1	Computer Science	Engineering	500000.00	CS301	Algo…	1	3	2
15	8	Sophia Wilson	1	2021	90	2	1	Computer Science	Engineering	500000.00	MATH101	Calc…	2	4	3

6.

```sql
SELECT
  Student.*,
  Course.*,
  Takes.grade,
  Instructor.*
FROM
  Student
LEFT JOIN
  Takes
  ON Takes.student_id = Student.student_id
LEFT JOIN
  Course
  ON Course.course_id = Takes.course_id
LEFT JOIN
  Instructor
  ON Instructor.instructor_id = Course.instructor_id;

```

Auto Commit | Manual

	student_id	name	dept_id	enrollment_year	total_credits	advisor_id	course_id	title	dept_id	credits	instructor_id	grade	instructor_id	name	dept_id	salary	hire_date
1	1	John Anderson	1	2022	30	1	CS101	Introduction to Programming	1	4	1	A	1	Dr. Alice Smith	1	85000.00	2018-05-15
2	1	John Anderson	1	2022	30	1	CS201	Data Structures	1	3	2	B+	2	Dr. Bob Johnson	1	92000.00	2015-08-20
3	1	John Anderson	1	2022	30	1	MATH101	Calculus I	2	4	3	A-	3	Dr. Carol Williams	2	78000.00	2020-01-10
4	2	Sarah Parker	1	2022	28	1	CS101	Introduction to Programming	1	4	1	B	1	Dr. Alice Smith	1	85000.00	2018-05-15
5	2	Sarah Parker	1	2022	28	1	CS301	Algorithms	1	3	2	A	2	Dr. Bob Johnson	1	92000.00	2015-08-20
6	3	Michael Chen	2	2021	75	3	MATH101	Calculus I	2	4	3	B+	3	Dr. Carol Williams	2	78000.00	2020-01-10
7	3	Michael Chen	2	2021	75	3	MATH201	Linear Algebra	2	3	(NULL)	A-	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
8	4	Jessica Taylor	3	2023	15	4	PHYS101	Mechanics	3	4	4	B+	4	Dr. David Brown	3	81000.00	2019-03-22
9	5	Kevin Rodriguez	(NULL)	2022	20	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
10	6	Olivia Martinez	4	2023	12	5	BIOL101	Biology I	4	4	5	A	5	Dr. Emma Davis	4	76000.00	2021-11-05
11	7	Daniel Kim	5	2022	45	7	CHEM101	General Chemistry	5	4	7	B	7	Dr. Frank Wilson	5	72000.00	2022-02-14
12	8	Sophia Wilson	1	2021	90	2	CS101	Introduction to Programming	1	4	1	A	1	Dr. Alice Smith	1	85000.00	2018-05-15
13	8	Sophia Wilson	1	2021	90	2	CS201	Data Structures	1	3	2	A	2	Dr. Bob Johnson	1	92000.00	2015-08-20
14	8	Sophia Wilson	1	2021	90	2	CS301	Algorithms	1	3	2	A-	2	Dr. Bob Johnson	1	92000.00	2015-08-20
15	8	Sophia Wilson	1	2021	90	2	MATH101	Calculus I	2	4	3	B+	3	Dr. Carol Williams	2	78000.00	2020-01-10