

Assignment – Student Information System (SISDB) Output Screenshots

Task 1

Students table

	student_id	first_name	last_name	date_of_birth	email	phone_number
▶	110	Hughie	Campbell	1995-08-15	hughie.campbell@gmail.com	9837567890
	210	Billy	Butcher	1998-06-20	billy.butcher@gmail.com	7345678901
	310	Albert	Bosh	2000-01-10	albert@gmail.com	9456789012
	410	Meera	Patil	1997-05-25	meera13@gmail.com	8567890123
	510	David	Williams	1999-09-30	david54@gmail.com	7878901234
	610	Emma	Jones	1996-12-14	emma.jones@gmail.com	8789012345
	710	Franklin	Miller	1994-07-07	franklinmiller@gmail.com	7890123456
	810	Calvin	Davis	2001-03-11	calvindavis@gmail.com	8901234567
	910	Lohith	Aksha	1993-11-22	lohithaksha12@gmail.com	9012345678
	1010	Isaac	Newton	1998-04-05	isaacnetwon100@gmail.com	8123456789
★	NULL	NULL	NULL	NULL	NULL	NULL

Courses table

	course_id	course_name	credits	teacher_id
▶	C01	Artificial Intelligence	4	1
	C02	Internet of Things	3	2
	C03	Computer Science	4	3
	C04	Cyber Security	3	4
	C05	Data Science	4	5
	C06	Software Engineering	3	6
	C07	Blockchain Technology	4	7
	C08	Cloud Computing	3	8
	C09	Machine Learning	4	9
	C10	Embedded Systems	3	10
★	NULL	NULL	NULL	NULL

Teacher table

	teacher_id	first_name	last_name	email
▶	1	Reachel	Merlin	reachelmerlin@gmail.com
	2	Aashwa	Damin	aashwadamin@gmail.com
	3	Moshina	Rahmath	moshina.rahmath@gmail.com
	4	Naveen	Gupte	Naveen13@gmail.com
	5	Kalai	Sudar	kalaisudar46@gmail.com
	6	Barath	Kumar	barathkumar@gmail.com
	7	Suha	shre	suha.shre@gmail.com
	8	Jana	Varshini	jana.varshini@gmail.com
	9	Felci	Christina	felci10christina@gmail.com
	10	Evangelin	Jenifer	evejeni23@gmail.com
★	NULL	NULL	NULL	NULL

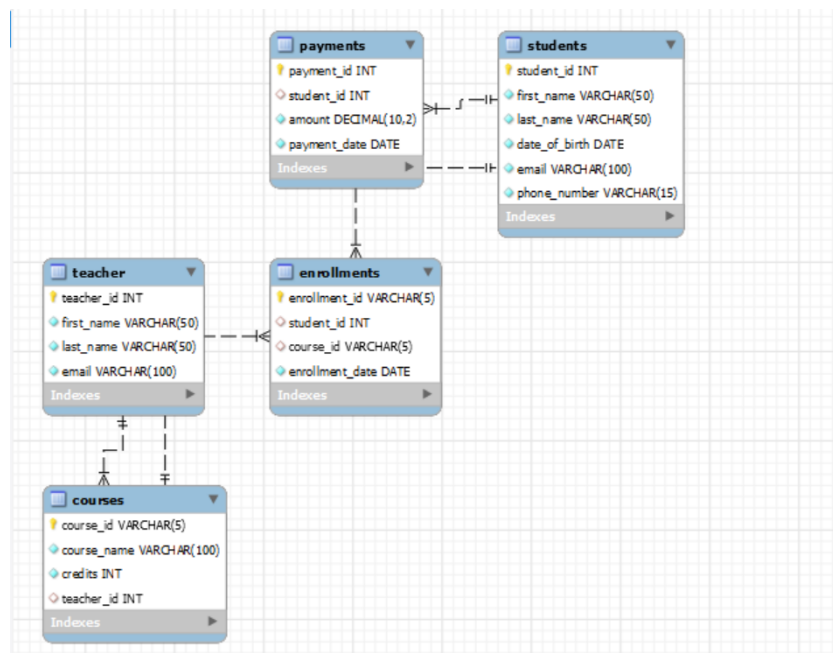
Enrollments table

	enrollment_id	student_id	course_id	enrollment_date
▶	E01	110	C01	2024-01-10
	E02	210	C02	2024-01-12
	E03	310	C03	2024-01-14
	E04	410	C04	2024-01-16
	E05	510	C05	2024-01-18
	E06	610	C06	2024-01-20
	E07	710	C07	2024-01-22
	E08	810	C08	2024-01-24
	E09	910	C09	2024-01-26
	E10	1010	C10	2024-01-28
✱	NULL	NULL	NULL	NULL

Payments table

	payment_id	student_id	amount	payment_date
▶	11	110	500.00	2024-02-10
	22	210	600.00	2024-02-12
	33	310	450.00	2024-02-14
	44	410	700.00	2024-02-16
	55	510	550.00	2024-02-18
	66	610	650.00	2024-02-20
	77	710	500.00	2024-02-22
	88	810	400.00	2024-02-24
	99	910	300.00	2024-02-26
	100	1010	750.00	2024-02-28
✱	NULL	NULL	NULL	NULL

Er diagram



Task 2 - Select, Where, Between, AND, LIKE:

1) insert into Students (student_id,first_name, last_name, date_of_birth, email, phone_number) values(1011, 'Firthose', 'Banu', '1995-08-15', 'firthosebanu@example.com', '7564667890');

	student_id	first_name	last_name	date_of_birth	email	phone_number
▶	110	Hughie	Campbell	1995-08-15	hughie.campbell@gmail.com	9837567890
	210	Billy	Butcher	1998-06-20	billy.butcher@gmail.com	7345678901
	310	Albert	Bosh	2000-01-10	albert@gmail.com	9456789012
	410	Meera	Patil	1997-05-25	meera13@gmail.com	8567890123
	510	David	Williams	1999-09-30	david54@gmail.com	7878901234
	610	Emma	Jones	1996-12-14	emma.jones@gmail.com	8789012345
	710	Franklin	Miller	1994-07-07	franklinmiller@gmail.com	7890123456
	810	Calvin	Davis	2001-03-11	calvindavis@gmail.com	8901234567
	910	Lohith	Aksha	1993-11-22	lohithaksha12@gmail.com	9012345678
	1010	Isaac	Newton	1998-04-05	isaacnetwon100@gmail.com	8123456789
	1011	Firthose	Banu	1995-08-15	firthosebanu@example.com	7564667890
*	NULL	NULL	NULL	NULL	NULL	NULL

2) insert into Enrollments (enrollment_id, student_id, course_id, enrollment_date) values ('E11', 1011, 'C01', '2024-03-01');

	enrollment_id	student_id	course_id	enrollment_date
▶	E01	110	C01	2024-01-10
	E02	210	C02	2024-01-12
	E03	310	C03	2024-01-14
	E04	410	C04	2024-01-16
	E05	510	C05	2024-01-18
	E06	610	C06	2024-01-20
	E07	710	C07	2024-01-22
	E08	810	C08	2024-01-24
	E09	910	C09	2024-01-26
	E10	1010	C10	2024-01-28
	E11	1011	C01	2024-03-01
*	NULL	NULL	NULL	NULL


3) update Teacher set email = 'hello.reachelmerlin@gmail.com'

WHERE teacher_id = 1;

	teacher_id	first_name	last_name	email
▶	1	Reachel	Merlin	hello.reachelmerlin@gmail.com
	2	Aashwa	Damin	aashwadamin@gmail.com
	3	Moshina	Rahmath	moshina.rahmath@gmail.com
	4	Naveen	Gupte	Naveen13@gmail.com
	5	Kalai	Sudar	kalaisudar46@gmail.com
	6	Barath	Kumar	barathkumar@gmail.com
	7	Suha	shre	suha.shre@gmail.com
	8	Jana	Varshini	jana.varshini@gmail.com
	9	Felci	Christina	felci10christina@gmail.com
	10	Evangelin	Jenifer	evejeni23@gmail.com
*	NULL	NULL	NULL	NULL


4) delete from Enrollments where student_id = 110 and course_id = 'C01';

Result Grid



Filter Rows:

Edit:



	enrollment_id	student_id	course_id	enrollment_date
▶	E02	210	C02	2024-01-12
	E03	310	C03	2024-01-14
	E04	410	C04	2024-01-16
	E05	510	C05	2024-01-18
	E06	610	C06	2024-01-20
	E07	710	C07	2024-01-22
	E08	810	C08	2024-01-24
	E09	910	C09	2024-01-26
	E10	1010	C10	2024-01-28
	E11	1011	C01	2024-03-01

5) update Courses set teacher_id = 2 where course_id = 'C09';

course_id	course_name	credits	teacher_id
C01	Artificial Intelligence	4	1
C02	Internet of Things	3	2
C03	Computer Science	4	3
C04	Cyber Security	3	4
C05	Data Science	4	5
C06	Software Engineering	3	6
C07	Blockchain Technology	4	7
C08	Cloud Computing	3	8
C09	Machine Learning	4	2
C10	Embedded Systems	3	10

6) Delete a specific student and remove their enrollments (Deleting Student ID 210)

delete from Payments where student_id = 210;

delete from Enrollments where student_id = 210;

delete from Students where student_id = 210;

student_id	first_name	last_name	date_of_birth	email	phone_number
110	Hughie	Campbell	1995-08-15	hughie.campbell@gmail.com	9837567890
310	Albert	Bosh	2000-01-10	albert@gmail.com	9456789012
410	Meera	Patil	1997-05-25	meera13@gmail.com	8567890123
510	David	Williams	1999-09-30	david54@gmail.com	7878901234
610	Emma	Jones	1996-12-14	emma.jones@gmail.com	8789012345
710	Franklin	Miller	1994-07-07	franklinmiller@gmail.com	7890123456
810	Calvin	Davis	2001-03-11	calvindavis@gmail.com	8901234567
910	Lohith	Aksha	1993-11-22	lohithaksha12@gmail.com	9012345678
1010	Isaac	Newton	1998-04-05	isaacnetwon100@gmail.com	8123456789
1011	Firthose	Banu	1995-08-15	firthosebanu@example.com	7564667890

7) update Payments set amount = 750.00 where payment_id = 22;

	payment_id	student_id	amount	payment_date
▶	11	110	500.00	2024-02-10
	33	310	450.00	2024-02-14
	44	410	700.00	2024-02-16
	55	510	550.00	2024-02-18
	66	610	650.00	2024-02-20
	77	710	500.00	2024-02-22
	88	810	400.00	2024-02-24
	99	910	750.00	2024-02-26
	100	1010	750.00	2024-02-28
★	NULL	NULL	NULL	NULL

Task 3. Aggregate functions, Having, Order By, GroupBy and Joins:

- 1) select s.student_id, s.first_name, s.last_name, sum(p.amount) as total_payment from students s join payments p on s.student_id = p.student_id where s.student_id = 110 group by s.student_id, s.first_name, s.last_name;

	student_id	first_name	last_name	total_payment
▶	110	Hughie	Campbell	500.00

- 2) select c.course_id, c.course_name, count(e.student_id) as student_count from courses c left join enrollments e on c.course_id = e.course_id group by c.course_id, c.course_name order by student_count desc;

	course_id	course_name	student_count
▶	C01	Artificial Intelligence	1
	C03	Computer Science	1
	C04	Cyber Security	1
	C05	Data Science	1
	C06	Software Engineering	1
	C07	Blockchain Technology	1
	C08	Cloud Computing	1
	C09	Machine Learning	1
	C10	Embedded Systems	1
	C02	Internet of Things	0

- 3) select s.student_id, s.first_name, s.last_name from students s left join enrollments e on s.student_id = e.student_id where e.student_id is null;

	student_id	first_name	last_name
▶	110	Hughie	Campbell

4) select s.first_name, s.last_name, c.course_name from students s join enrollments e on s.student_id = e.student_id join courses c on e.course_id = c.course_id;

	first_name	last_name	course_name
▶	Albert	Bosh	Computer Science
	Meera	Patil	Cyber Security
	David	Williams	Data Science
	Emma	Jones	Software Engineering
	Franklin	Miller	Blockchain Technology
	Calvin	Davis	Cloud Computing
	Lohith	Aksha	Machine Learning
	Isaac	Newton	Embedded Systems
	Firthose	Banu	Artificial Intelligence

5) SELECT t.first_name, t.last_name, c.course_name FROM Teacher t JOIN Courses c ON t.teacher_id = c.teacher_id;

	first_name	last_name	course_name
▶	Reachel	Merlin	Artificial Intelligence
	Aashwa	Damin	Internet of Things
	Aashwa	Damin	Machine Learning
	Moshina	Rahmath	Computer Science
	Naveen	Gupte	Cyber Security
	Kalai	Sudar	Data Science
	Barath	Kumar	Software Engineering
	Suha	shre	Blockchain Technology
	Jana	Varshini	Cloud Computing
	Evangelin	Jenifer	Embedded Systems

6) select s.first_name, s.last_name, e.enrollment_date from students s join enrollments e on s.student_id = e.student_id join courses c on e.course_id = c.course_id where c.course_name = 'artificial intelligence' order by e.enrollment_date;

	first_name	last_name	enrollment_date
▶	Firthose	Banu	2024-03-01

7) select s.student_id, s.first_name, s.last_name from students s left join payments p on s.student_id = p.student_id where p.student_id is null;

Result Grid	Filter Rows:	Export
student_id	first_name	last_name
1011	Firthose	Banu

8) select c.course_id, c.course_name from courses c left join enrollments e on c.course_id = e.course_id where e.course_id is null;

course_id	course_name
C02	Internet of Things

9) select e.student_id, s.first_name, s.last_name, count(e.course_id) as num_of_courses from enrollments e join students s on e.student_id = s.student_id group by e.student_id, s.first_name, s.last_name having num_of_courses > 1;

Result Grid

Filter Rows:

Export:

student_id	first_name	last_name	num_of_courses
------------	------------	-----------	----------------

10) select t.teacher_id, t.first_name, t.last_name from teacher t left join courses c on t.teacher_id = c.teacher_id where c.teacher_id is null;

teacher_id	first_name	last_name
9	Felci	Christina

Task 4. Subquery and its type:

1) select avg(student_count) as avg_students_per_course from (select course_id, count(student_id) as student_count from enrollments group by course_id) as course_enrollments;

avg_students_per_course
1.0000

2) select student_id, first_name, last_name from students where student_id in (select student_id from payments where amount = (select max(amount) from payments));

	student_id	first_name	last_name
▶	910	Lohith	Aksha
	1010	Isaac	Newton
*	NULL	NULL	NULL

3) select avg(student_count) as average_enrollment from (select course_id, count(student_id) as student_count from enrollments group by course_id) as course_enrollments;

	course_id	course_name	enrollment_count
▶	C03	Computer Science	1
	C04	Cyber Security	1
	C05	Data Science	1
	C06	Software Engineering	1
	C07	Blockchain Technology	1
	C08	Cloud Computing	1
	C09	Machine Learning	1
	C10	Embedded Systems	1
	C01	Artificial Intelligence	1

4) select t.teacher_id, t.first_name, t.last_name, (select sum(p.amount) from payments p join enrollments e on p.student_id = e.student_id join courses c on e.course_id = c.course_id where c.teacher_id = t.teacher_id) as total_payments from teacher t;

	teacher_id	first_name	last_name	total_payments
▶	1	Reachel	Merlin	NULL
	2	Aashwa	Damin	750.00
	3	Moshina	Rahmath	450.00
	4	Naveen	Gupte	700.00
	5	Kalai	Sudar	550.00
	6	Barath	Kumar	650.00
	7	Suha	shre	500.00
	8	Jana	Varshini	400.00
	9	Felci	Christina	NULL
	10	Evangelin	Jenifer	750.00

5) select s.student_id, s.first_name, s.last_name from students s where (select count(distinct course_id) from enrollments where student_id = s.student_id) = (select count(*) from courses);

	student_id	first_name	last_name
*	NULL	NULL	NULL

6) select teacher_id, first_name, last_name from teacher where teacher_id not in (select distinct teacher_id from courses);

	teacher_id	first_name	last_name
▶	9	Felci	Christina
*	NULL	NULL	NULL

7) select avg(age) as average_age from (select datediff(curdate(), date_of_birth) / 365 as age from students) as age_subquery;

	average_age
▶	27.91616438

8) select course_id, course_name from courses where course_id not in (select distinct course_id from enrollments);

	course_id	course_name
▶	C02	Internet of Things
*	NULL	NULL

9) select s.student_id, s.first_name, s.last_name, c.course_name, (select sum(p.amount) from payments p where p.student_id = s.student_id) as total_payment from students s join enrollments e on s.student_id = e.student_id join courses c on e.course_id = c.course_id;

	student_id	first_name	last_name	course_name	total_payment
▶	310	Albert	Bosh	Computer Science	450.00
	410	Meera	Patil	Cyber Security	700.00
	510	David	Williams	Data Science	550.00
	610	Emma	Jones	Software Engineering	650.00
	710	Franklin	Miller	Blockchain Technology	500.00
	810	Calvin	Davis	Cloud Computing	400.00
	910	Lohith	Aksha	Machine Learning	750.00
	1010	Isaac	Newton	Embedded Systems	750.00
	1011	Firthose	Banu	Artificial Intelligence	NULL

10) select student_id, first_name, last_name from students where student_id in select student_id from payments group by student_id having count(payment_id) > 1);

	student_id	first_name	last_name
*	NULL	NULL	NULL

11) select s.student_id, s.first_name, s.last_name, sum(p.amount) as total_payments from students s join payments p on s.student_id = p.student_id group by s.student_id;

	student_id	first_name	last_name	total_payments
▶	110	Hughie	Campbell	500.00
	310	Albert	Bosh	450.00
	410	Meera	Patil	700.00
	510	David	Williams	550.00
	610	Emma	Jones	650.00
	710	Franklin	Miller	500.00
	810	Calvin	Davis	400.00
	910	Lohith	Aksha	750.00
	1010	Isaac	Newton	750.00

12) select c.course_name, count(e.student_id) as student_count from courses c left join enrollments e on c.course_id = e.course_id group by c.course_name;

	course_name	student_count
▶	Artificial Intelligence	1
	Internet of Things	0
	Computer Science	1
	Cyber Security	1
	Data Science	1
	Software Engineering	1
	Blockchain Technology	1
	Cloud Computing	1
	Machine Learning	1
	Embedded Systems	1

13) select s.student_id, s.first_name, s.last_name, avg(p.amount) as avg_payment from students join payments p on s.student_id = p.student_id group by s.student_id;

	student_id	first_name	last_name	avg_payment
▶	110	Hughie	Campbell	500.000000
	310	Albert	Bosh	450.000000
	410	Meera	Patil	700.000000
	510	David	Williams	550.000000
	610	Emma	Jones	650.000000
	710	Franklin	Miller	500.000000
	810	Calvin	Davis	400.000000
	910	Lohith	Aksha	750.000000
	1010	Isaac	Newton	750.000000