GROUP - 4 - INTELLITECH
Members: Bantilo, Jade Daniele M.
Corda, Ryan P.
Derige, Paul Angelo
Eugenio, Shiloh B.

Scenario: Intellitech Apartments You have completed the designs for the apartment management database. You reviewed it and all the business rules with the owners and they are eager to proceed. Now you need to take your design and translate it into an actual database. Once you have done that you know that you will need to enter data to test the database, to make sure it does, in fact, store all the required data.

1. Review your diagram for the database making sure that the design is complete and normalized.

2. Create the database in SQL Server

```
1 CREATE DATABASE intellitech_apartments;
```

3. Create the tables in the new database, selecting appropriate data types for the columns, setting a primary key for each table, and setting allow nulls as appropriate

buildings table

```
1 CREATE TABLE buildings (
2     id INT(11) AUTO_INCREMENT PRIMARY KEY,
3     name VARCHAR(50) NOT NULL,
4     address VARCHAR(100) NOT NULL,
5     num_floors INT(11) NOT NULL,
6     created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
7     updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP
8 );
```

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐	1	id	int(11)			No	*None*		AUTO_INCREMENT	Change Drop More
☐	2	name	varchar(50)	utf8mb4_general_ci		No	*None*			Change Drop More
☐	3	address	varchar(100)	utf8mb4_general_ci		No	*None*			Change Drop More
☐	4	num_floors	int(11)			No	*None*			Change Drop More
☐	5	created_at	timestamp			No	current_timestamp()			Change Drop More
☐	6	updated_at	timestamp			No	current_timestamp()		ON UPDATE CURRENT_TIMESTAMP()	Change Drop More

apartments table

```sql
CREATE TABLE apartments (
    id INT(11) AUTO_INCREMENT PRIMARY KEY,
    unit_number VARCHAR(10) NOT NULL,
    floor INT(11) NOT NULL,
    building_id INT(11) NOT NULL,
    num_bedrooms INT(11) NOT NULL,
    num_bathrooms INT(11) NOT NULL,
    square_footage DECIMAL(10,2),
    rent DECIMAL(10,2) NOT NULL,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
    FOREIGN KEY (building_id) REFERENCES buildings(id)
);
```

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	unit_number	varchar(10)	utf8mb4_general_ci		No	None			Change Drop More
3	floor	int(11)			No	None			Change Drop More
4	building_id	int(11)			No	None			Change Drop More
5	num_bedrooms	int(11)			No	None			Change Drop More
6	num_bathrooms	int(11)			No	None			Change Drop More
7	square_footage	decimal(10,2)			Yes	NULL			Change Drop More
8	rent	decimal(10,2)			No	None			Change Drop More
9	created_at	timestamp			No	current_timestamp()			Change Drop More
10	updated_at	timestamp			No	current_timestamp()		ON UPDATE CURRENT_TIMESTAMP()	Change Drop More

tenants table

```sql
CREATE TABLE tenants (
    id INT(11) AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(50) NOT NULL,
    phone VARCHAR(15),
    email VARCHAR(50),
    apartment_id INT(11),
    FOREIGN KEY (apartment_id) REFERENCES apartments(id)
);
```

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	name	varchar(50)	utf8mb4_general_ci		No	None			Change Drop More
3	phone	varchar(15)	utf8mb4_general_ci		Yes	NULL			Change Drop More
4	email	varchar(50)	utf8mb4_general_ci		Yes	NULL			Change Drop More
5	apartment_id	int(11)			Yes	NULL			Change Drop More

leases table

```sql
CREATE TABLE leases (
    id INT(11) AUTO_INCREMENT PRIMARY KEY,
    tenant_id INT(11) NOT NULL,
    apartment_id INT(11) NOT NULL,
    start_date DATE NOT NULL,
    end_date DATE NOT NULL,
    amount DECIMAL(10,2) NOT NULL,
    FOREIGN KEY (tenant_id) REFERENCES tenants(id),
    FOREIGN KEY (apartment_id) REFERENCES apartments(id)
);
```

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	tenant_id	int(11)			No	None			Change Drop More
3	apartment_id	int(11)			No	None			Change Drop More
4	start_date	date			No	None			Change Drop More
5	end_date	date			No	None			Change Drop More
6	amount	decimal(10,2)			No	None			Change Drop More

maintenance_requests

```sql
CREATE TABLE maintenance_requests (
    id INT(11) AUTO_INCREMENT PRIMARY KEY,
    apartment_id INT(11) NOT NULL,
    tenant_id INT(11) NOT NULL,
    request_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    issue_description TEXT,
    status ENUM('Pending', 'In Progress', 'Resolved') DEFAULT 'Pending',
    FOREIGN KEY (apartment_id) REFERENCES apartments(id),
    FOREIGN KEY (tenant_id) REFERENCES tenants(id)
);
```

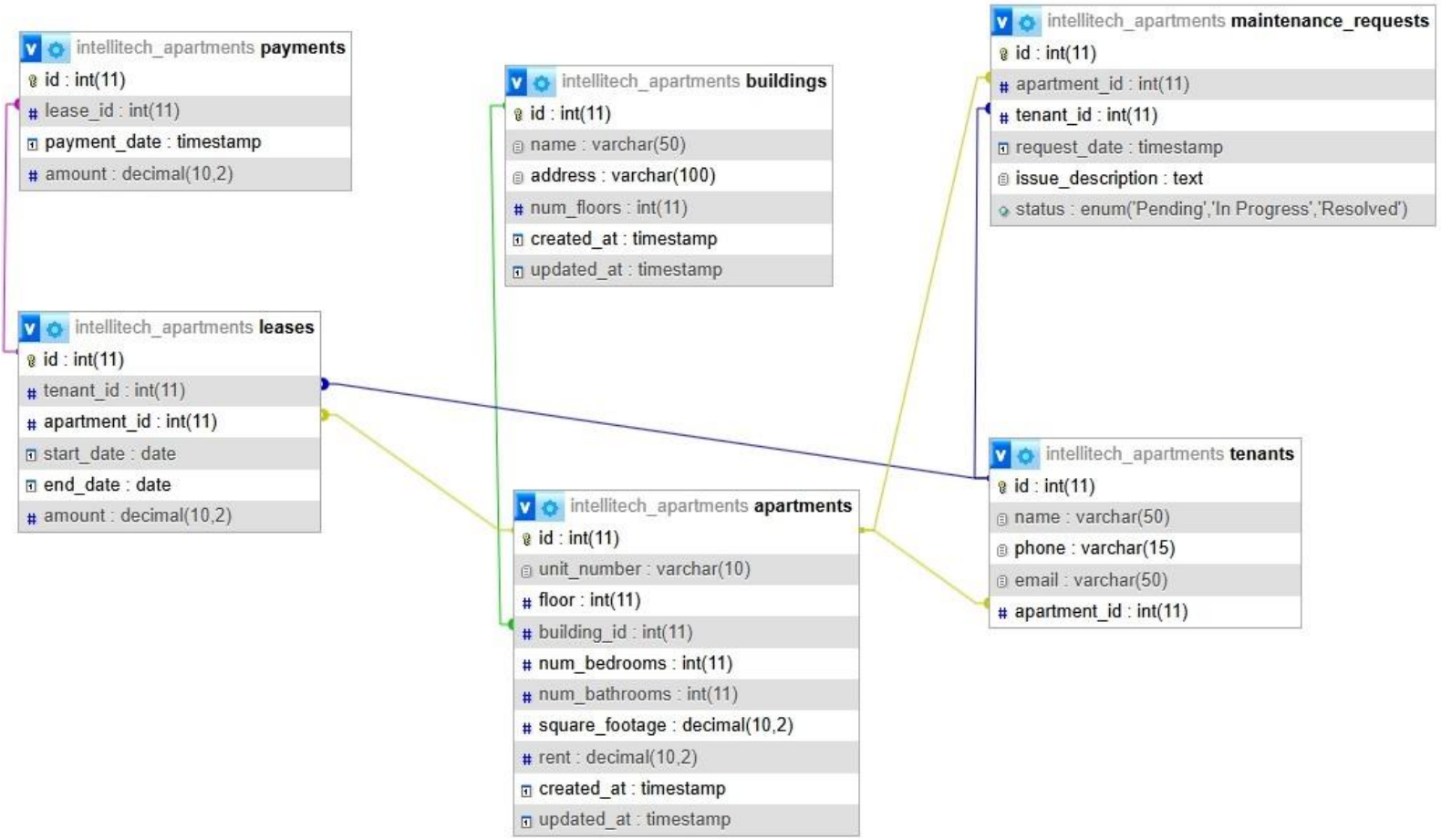
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	apartment_id	int(11)			No	None			Change Drop More
3	tenant_id	int(11)			No	None			Change Drop More
4	request_date	timestamp			No	current_timestamp()			Change Drop More
5	issue_description	text	utf8mb4_general_ci		Yes	NULL			Change Drop More
6	status	enum('Pending', 'In Progress', 'Resolved')	utf8mb4_general_ci		Yes	Pending			Change Drop More

payments table

```sql
CREATE TABLE payments (
    id INT(11) AUTO_INCREMENT PRIMARY KEY,
    lease_id INT(11) NOT NULL,
    payment_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    amount DECIMAL(10,2) NOT NULL,
    FOREIGN KEY (lease_id) REFERENCES leases(id)
);
```

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	lease_id	int(11)			No	None			Change Drop More
3	payment_date	timestamp			No	current_timestamp()			Change Drop More
4	amount	decimal(10,2)			No	None			Change Drop More

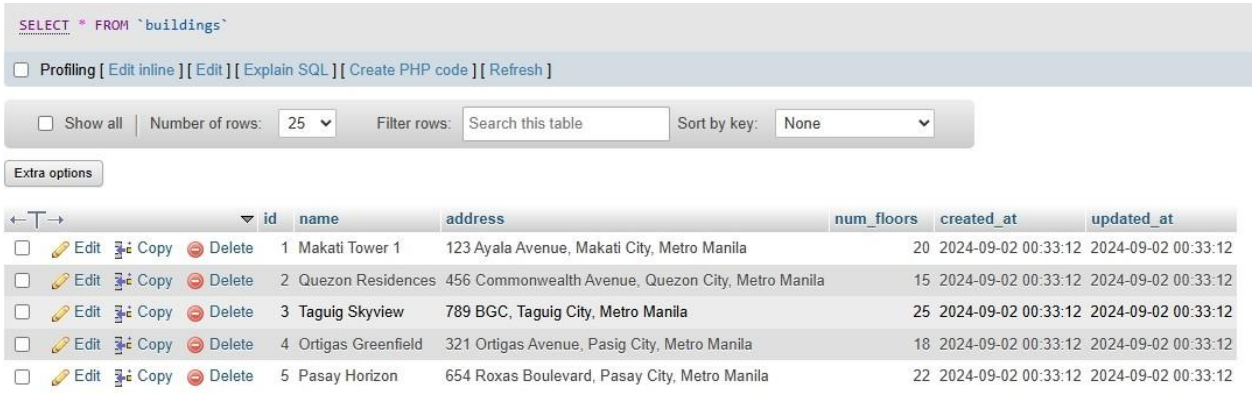
4. Create a database diagram and create the relationships among tables



5. Add some sample data to each table

Insert data in table buildings

```
INSERT INTO buildings (name, address, num_floors) VALUES
('Makati Tower 1', '123 Ayala Avenue, Makati City, Metro Manila', 20),
('Quezon Residences', '456 Commonwealth Avenue, Quezon City, Metro Manila', 15),
('Taguig Skyview', '789 BGC, Taguig City, Metro Manila', 25),
('Ortigas Greenfield', '321 Ortigas Avenue, Pasig City, Metro Manila', 18),
('Pasay Horizon', '654 Roxas Boulevard, Pasay City, Metro Manila', 22);
```



```
SELECT * FROM `buildings`
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

Extra options

	id	name	address	num_floors	created_at	updated_at
Edit Copy Delete	1	Makati Tower 1	123 Ayala Avenue, Makati City, Metro Manila	20	2024-09-02 00:33:12	2024-09-02 00:33:12
Edit Copy Delete	2	Quezon Residences	456 Commonwealth Avenue, Quezon City, Metro Manila	15	2024-09-02 00:33:12	2024-09-02 00:33:12
Edit Copy Delete	3	Taguig Skyview	789 BGC, Taguig City, Metro Manila	25	2024-09-02 00:33:12	2024-09-02 00:33:12
Edit Copy Delete	4	Ortigas Greenfield	321 Ortigas Avenue, Pasig City, Metro Manila	18	2024-09-02 00:33:12	2024-09-02 00:33:12
Edit Copy Delete	5	Pasay Horizon	654 Roxas Boulevard, Pasay City, Metro Manila	22	2024-09-02 00:33:12	2024-09-02 00:33:12

Insert data in table apartments

```
INSERT INTO apartments (unit_number, floor, building_id, num_bedrooms, num_bathrooms, square_footage, rent) VALUES
('101A', 1, 1, 2, 1, 45.50, 25000.00),
('202B', 2, 2, 3, 2, 60.00, 32000.00),
('303C', 3, 3, 1, 1, 35.75, 18000.00),
('404D', 4, 4, 3, 2, 70.25, 35000.00),
('505E', 5, 5, 2, 1, 50.00, 28000.00);
```

```
SELECT * FROM `apartments`
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

Extra options

	id	unit_number	floor	building_id	num_bedrooms	num_bathrooms	square_footage	rent	created_at	updated_at
Edit Copy Delete	1	101A	1	1	2	1	45.50	25000.00	2024-09-02 00:34:02	2024-09-02 00:34:02
Edit Copy Delete	2	202B	2	2	3	2	60.00	32000.00	2024-09-02 00:34:02	2024-09-02 00:34:02
Edit Copy Delete	3	303C	3	3	1	1	35.75	18000.00	2024-09-02 00:34:02	2024-09-02 00:34:02
Edit Copy Delete	4	404D	4	4	3	2	70.25	35000.00	2024-09-02 00:34:02	2024-09-02 00:34:02
Edit Copy Delete	5	505E	5	5	2	1	50.00	28000.00	2024-09-02 00:34:02	2024-09-02 00:34:02

Insert data in table tenants

```
INSERT INTO tenants (name, phone, email, apartment_id) VALUES
('Shiloh Eugenio', '09171234567', 'shiloeugenio@gmail.com', 1),
('Ryan Corda', '09182345678', 'ryancordao@gmail.com', 2),
('Paul Angelo Derige', '09223456789', 'paul_angelo@gmail.com', 3),
('Jade Daniele Bantilo', '09334567890', 'jadebantilo123@gmail.com', 4),
('Arriane Camille Pamintuan', '09445678901', 'camille_arriane@gmail.com', 5);
```

```
SELECT * FROM `tenants`
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

Extra options

	id	name	phone	email	apartment_id
Edit Copy Delete	1	Shiloh Eugenio	09171234567	shiloeugenio@gmail.com	1
Edit Copy Delete	2	Ryan Corda	09182345678	ryancordao@gmail.com	2
Edit Copy Delete	3	Paul Angelo Derige	09223456789	paul_angelo@gmail.com	3
Edit Copy Delete	4	Jade Daniele Bantilo	09334567890	jadebantilo123@gmail.com	4
Edit Copy Delete	5	Arriane Camille Pamintuan	09445678901	camille_arriane@gmail.com	5

Insert data in table leases

```
INSERT INTO leases (tenant_id, apartment_id, start_date, end_date, amount) VALUES
(1, 1, '2023-01-01', '2023-12-31', 25000.00),
(2, 2, '2023-01-01', '2023-12-31', 32000.00),
(3, 3, '2023-01-01', '2023-12-31', 18000.00),
(4, 4, '2023-01-01', '2023-12-31', 35000.00),
(5, 5, '2023-01-01', '2023-12-31', 28000.00);
```

```
SELECT * FROM `leases`
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

Extra options

	id	tenant_id	apartment_id	start_date	end_date	amount
Edit Copy Delete	1	1	1	2023-01-01	2023-12-31	25000.00
Edit Copy Delete	2	2	2	2023-01-01	2023-12-31	32000.00
Edit Copy Delete	3	3	3	2023-01-01	2023-12-31	18000.00
Edit Copy Delete	4	4	4	2023-01-01	2023-12-31	35000.00
Edit Copy Delete	5	5	5	2023-01-01	2023-12-31	28000.00

Insert data in table maintenance_requests

```
INSERT INTO maintenance_requests (apartment_id, tenant_id, issue_description, status) VALUES
(1, 1, 'Leaky faucet in the kitchen.', 'Pending'),
(2, 2, 'Air conditioner not cooling properly.', 'In Progress'),
(3, 3, 'Broken window latch.', 'Pending'),
(4, 4, 'Malfunctioning elevator.', 'Resolved'),
(5, 5, 'Clogged bathroom sink.', 'Pending');
```

```
SELECT * FROM `maintenance_requests`
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

Extra options

	id	apartment_id	tenant_id	request_date	issue_description	status
Edit Copy Delete	1	1	1	2024-09-02 00:38:16	Leaky faucet in the kitchen.	Pending
Edit Copy Delete	2	2	2	2024-09-02 00:38:16	Air conditioner not cooling properly.	In Progress
Edit Copy Delete	3	3	3	2024-09-02 00:38:16	Broken window latch.	Pending
Edit Copy Delete	4	4	4	2024-09-02 00:38:16	Malfunctioning elevator.	Resolved
Edit Copy Delete	5	5	5	2024-09-02 00:38:16	Clogged bathroom sink.	Pending

Insert data in table payments

```
INSERT INTO payments (lease_id, amount) VALUES
(1, 25000.00),
(2, 32000.00),
(3, 18000.00),
(4, 35000.00),
(5, 28000.00);
```

```
SELECT * FROM `payments`
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

Extra options

	id	lease_id	payment_date	amount
Edit Copy Delete	1	1	2024-09-02 00:40:00	25000.00
Edit Copy Delete	2	2	2024-09-02 00:40:00	32000.00
Edit Copy Delete	3	3	2024-09-02 00:40:00	18000.00
Edit Copy Delete	4	4	2024-09-02 00:40:00	35000.00
Edit Copy Delete	5	5	2024-09-02 00:40:00	28000.00

6. Documentation: Make a Data Dictionary that lists each table, all the columns for that table, the data types for each column.

apartments table

Column	Date Type	Description
id	int(11)	Primary key, auto-increment
unit_number	varchar(10)	Unit number
floor	int(11)	Floor number
building_id	int(11)	Foreign key, references buildings(id)
num_bedrooms	int(11)	Number of bedrooms
num_bathrooms	int(11)	Number of bathrooms
square_footage	decimal(10, 2)	Square footage of the apartment (nullable)
rent	decimal(10, 2)	Monthly rent amount
created_at	timestamp	Timestamp when the record was created (default: current timestamp)
updated_at	timestamp	Timestamp when the record was last updated (default: current timestamp, updated on change)

buildings table

Column	Data Type	Description
id	int(11)	Primary key, auto-increment
name	varchar(50)	Building name
address	varchar(100)	Building address
num_floors	int(11)	Number of floors in the building
created_at	timestamp	Timestamp when the record was created (default: current timestamp)
updated_at	timestamp	Timestamp when the record was last updated (default: current timestamp, updated on change)

leases table

Column	Type	Description
id	int(11)	Primary key, auto-increment
tenant_id	int(11)	Foreign key, references tenants(id)
apartment_id	int(11)	Foreign key, references apartments(id)
start_date	date	Lease start date
end_date	date	Lease end date
amount	decimal(10, 2)	Lease amount

maintenance_requests table

Column	Data Type	Description
id	int(11)	Primary key, auto-increment
apartment_id	int(11)	Foreign key, references apartments(id)
tenant_id	int(11)	Foreign key, references tenants(id)
request_date	timestamp	Timestamp when the request was made (default: current timestamp)
issue_description	text	Description of the maintenance issue
status	enum('Pending', 'In Progress', 'Resolved')	Status of the request

payments table

Column	Data Type	Description
id	int(11)	Primary key, auto-increment
lease_id	int(11)	Foreign key, references leases(id)
payment_date	timestamp	Timestamp when the payment was made (default: current timestamp)
amount	decimal(10, 2)	Payment amount

tenants table

Column	Data Type	Description
id	int(11)	Primary key, auto-increment
name	varchar(50)	Tenant's name
phone	varchar(15)	Tenant's phone number (nullable)
email	varchar(50)	Tenant's email address (nullable)
apartment_id	int(11)	Foreign key, references apartments(id)