

REAL ESTATE BROKERAGE

Software Testing Sample Data & Test Scenarios

Dummy schema • Functional, Negative, Validation & Data Integrity testing

Document	Dummy Real Estate Brokerage QA Pack
Purpose	Manual software testing, demo, and test-case design
Currency	INR (Indian Rupees)
Scope	Agents, Clients, Properties, Deals, Commission Payments
Data	All records and contact details are fictional

Note: This document is an illustrative QA sample, not a production database specification or legal/financial guidance. Expected results assume the application follows the sample rules listed here. Real brokerage rules may vary.

1. Dummy Schema & Relationships

Agents manage property listings and deals. **Clients** may participate in deals. **Properties** can have historical deal records. **Deals** can have one or more commission payment records. PK = Primary Key; FK = Foreign Key.

Agents

Test records for agent management.

agent_id (PK)	name	email	status
AG001	Aarav Deshmukh	aarav@example.test	Active
AG002	Mira Kulkarni	mira@example.test	Active
AG003	Kabir Shah	kabir@example.test	Inactive

Clients

Test records for buyers, sellers, and tenants.

client_id (PK)	name	type	phone (dummy)	email
CL001	Neha Patil	Buyer	9000000201	neha@example.test
CL002	Rohan Mehta	Seller	9000000202	rohan@example.test
CL003	Sara Khan	Tenant	9000000203	sara@example.test

Properties

Asking price is in INR. Listed-by agent must exist.

property_id (PK)	property type	city	asking price	agent FK	status
PR001	2 BHK Apartment	Pune	8500000	AG001	Available
PR002	1 BHK Apartment	Mumbai	6200000	AG002	Under Offer
PR003	Office Unit	Navi Mumbai	12000000	AG001	Sold

Deals / Transactions

sale_price is an agreed transaction amount. Rental deal example uses monthly rent as the basis.

deal_id (PK)	property FK	client FK	agent FK	type	amount (INR)	date	status
DL001	PR003	CL002	AG001	Sale	11800000	2026-09-18	Completed
DL002	PR002	CL001	AG002	Sale	6100000	2026-10-02	In Progress
DL003	PR001	CL003	AG001	Rental	35000/month	2026-10-10	Pending

Commission Payments

For sale deals, sample commission = agreed sale price × rate. For rentals, this sample uses one month's rent.

payment_id	deal FK	rate / rule	commission due	paid amount	payment date	status
CP001	DL001	2%	236000	236000	2026-09-20	Paid
CP002	DL002	2%	122000	50000	2026-10-05	Part-paid
CP003	DL003	1 month's rent	35000	0	—	Unpaid

2. Suggested Test Environment & Preconditions

- Use a QA/staging environment, not live customer or property data.
- Load the dummy records above before executing tests.
- The tester has permission to create, view, edit, and delete records where applicable.
- Expected business rules: IDs are unique; required fields cannot be blank; referenced Agent, Client, Property, and Deal must exist.
- Commission cannot be negative; payment cannot be negative or exceed the outstanding commission without an approved adjustment.
- Dates must be valid dates; deal completion should require a valid agreed amount and required linked records.

3. Test Scenarios / Test Cases

Suggested manual test suite. Record Actual Result and Pass/Fail during execution. All IDs and values are illustrative.

TC ID	Module	Scenario	Test Steps / Input	Expected Result	Type	Actual / Status
TC01	Agent	Create agent with valid required fields	Enter unique ID AG004, name, valid email; Save	Agent is created and appears in list	Functional	
TC02	Agent	Duplicate agent ID	Create another agent using AG001	Save is blocked; duplicate ID message shown	Negative / Validation	
TC03	Agent	Required name missing	Leave name blank and Save	Name validation appears; no record created	Validation	
TC04	Client	Create client with valid data	Enter unique client ID and valid details	Client is saved and details display correctly	Functional	
TC05	Client	Invalid email format	Enter `name@` as email	Invalid email is rejected or flagged	Validation	
TC06	Property	Create valid property	Enter PR004, property type, city, price, AG001	Property is saved with correct agent link	Functional	
TC07	Property	Negative asking price	Enter asking price -1000	Save is blocked; price must be zero or greater	Negative / Validation	
TC08	Property	Non-existent listing agent	Set agent ID AG999	Save is blocked; agent reference error shown	Negative / Data integrity	
TC09	Property	Missing property type or city	Leave required field empty	Inline validation appears; record not saved	Validation	
TC10	Property	Search property by city	Search Pune	PR001 is shown; unrelated cities are not shown	Functional	
TC11	Property	Update listing status	Change PR001 from Available to Under Offer	Updated status is saved and shown consistently	Functional	

TC ID	Module	Scenario	Test Steps / Input	Expected Result	Type	Actual / St
TC12	Property	Delete property linked to a deal	Attempt to delete PR003 which is referenced by DL001	System prevents unsafe deletion or follows documented archive rule	Negative / Data integrity	
TC13	Deal	Create completed sale	Create deal with existing PR003, CL002, AG001, valid price/date	Deal is saved with correct relationships	Functional	
TC14	Deal	Unknown property ID	Use property ID PR999	Save is blocked; no orphan deal is created	Negative / Data integrity	
TC15	Deal	Unknown client or agent	Use CL999 or AG999	Save is blocked with useful validation	Negative / Data integrity	
TC16	Deal	Sale price missing for completed sale	Set status Completed and leave sale price blank	Save is blocked until required amount is entered	Validation	
TC17	Deal	Invalid date	Enter 2026-02-30	Invalid date is rejected; no incorrect date is stored	Validation	
TC18	Commission	Calculate 2% sale commission	For DL001, sale price 11800000 and rate 2%	Commission due equals 236000	Functional / Calculation	
TC19	Commission	Calculate commission for DL002	Sale price 6100000 and rate 2%	Commission due equals 122000	Functional / Calculation	
TC20	Commission	Negative commission rate	Enter -2%	Value is rejected; commission remains unchanged	Negative / Validation	
TC21	Payment	Record full commission payment	Record 236000 against DL001 commission due 236000	Payment saves; status becomes Paid; balance is 0	Functional	
TC22	Payment	Record partial payment	Record 50000 against DL002 due 122000	Payment saves; balance is 72000; status is Part-paid	Functional / Calculation	
TC23	Payment	Payment exceeds commission due	Attempt payment of 250000 against DL001 due 236000	Payment is blocked unless approved adjustment workflow permits it	Negative / Validation	
TC24	Payment	Negative payment amount	Enter -500	Save is blocked; no negative payment is recorded	Negative / Validation	
TC25	Payment	Duplicate payment submission	Click Save twice or resubmit same request	No duplicate payment is created; one payment record exists	Negative / Data integrity	
TC26	Access	Unauthorized user edits commission	Sign in as a read-only user and try to edit payment	Edit is unavailable or access is denied	Security / Negative	
TC27	List / UI	Refresh after adding a record	Create a new client, return to list, refresh	New client appears once and values persist	Functional	
TC28	Data integrity	Delete client linked to a deal	Attempt to delete CL002 used by DL001	System blocks deletion or archives client without breaking deal history	Negative / Data integrity	
TC29	Search	Search with no matching property	Search for a unique nonexistent city	Clear no-results state appears; page does not crash	Negative / UI	
TC30	Validation	Whitespace-only name	Enter spaces in required name field	Value is treated as blank and rejected	Validation	

4. Calculation Reference

Sample Expected Commission

All values in INR unless stated otherwise.

Deal	Type	Deal amount	Rate / rule	Due	Paid	Balance	Status
DL001	Sale	11,800,000	2%	236,000	236,000	0	Paid
DL002	Sale	6,100,000	2%	122,000	50,000	72,000	Part-paid
DL003	Rental	35,000 / month	1 month's rent	35,000	0	35,000	Unpaid

5. SQL Queries for Data Checks (MySQL)

```
-- Find properties with missing/invalid listing agent references
SELECT p.property_id, p.listed_by_agent_id
FROM properties p
LEFT JOIN agents a ON a.agent_id = p.listed_by_agent_id
WHERE p.listed_by_agent_id IS NOT NULL AND a.agent_id IS NULL;

-- Find deals whose referenced records are missing
SELECT d.deal_id
FROM deals d
LEFT JOIN properties p ON p.property_id = d.property_id
LEFT JOIN clients c ON c.client_id = d.client_id
LEFT JOIN agents a ON a.agent_id = d.agent_id
WHERE p.property_id IS NULL OR c.client_id IS NULL OR a.agent_id IS NULL;

-- Compare total paid to commission due (example schema)
SELECT deal_id, SUM(commission_due) AS due, SUM(amount_paid) AS paid
FROM commission_payments
GROUP BY deal_id
HAVING SUM(amount_paid) > SUM(commission_due);
```

6. Execution Result Template

Fill this in during testing

TC ID	Actual result	Pass / Fail / Blocked	Bug ID / Notes
TC01			
TC02			
TC03			
TC18			
TC21			
TC26			

7. Important Assumptions

- The test suite is based on the fictional schema and rules in this PDF; adjust it to match the actual application requirements.
- The example uses a flat 2% sale commission and one month's rent for rental brokerage solely to make expected results testable.
- This sample does not cover jurisdiction-specific taxes, contracts, GST, brokerage caps, escrow, or regulatory compliance.
- Use masked or synthetic contact information in test environments. The example uses reserved `.test` email domains and dummy phone values.