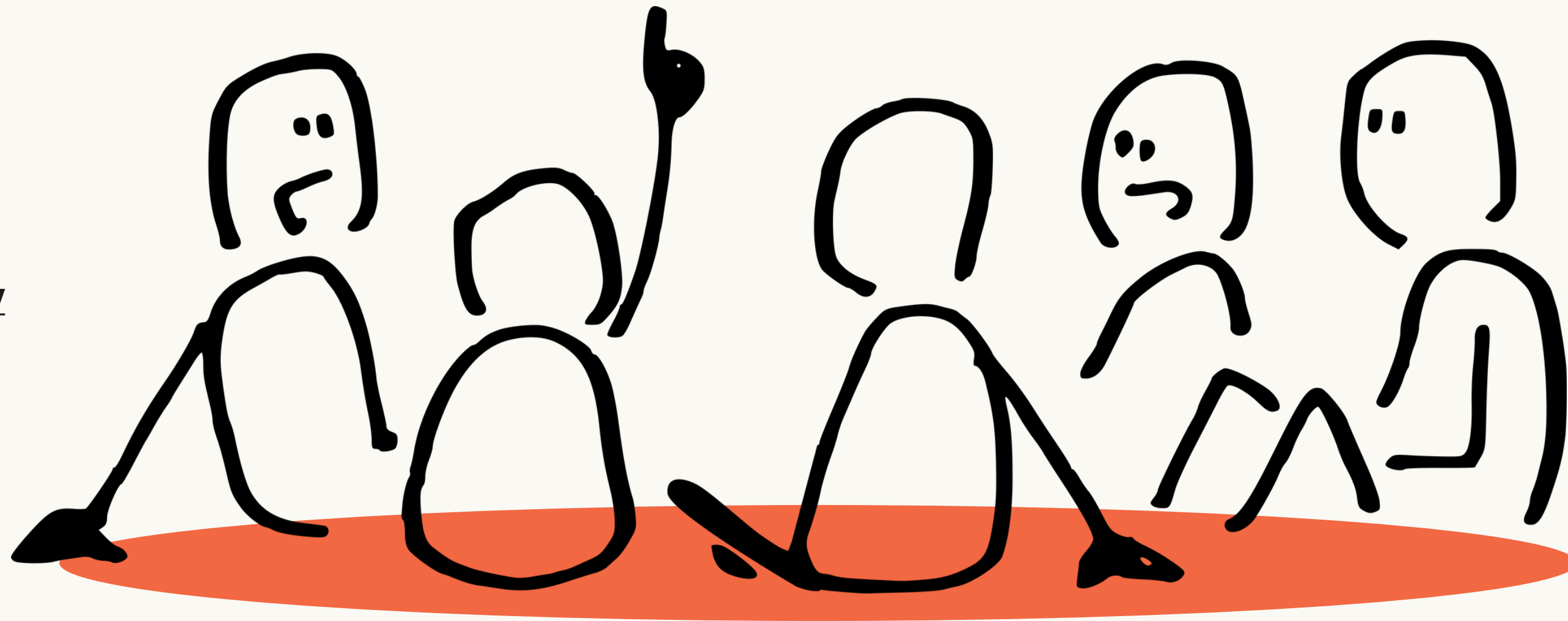


G-A-G

Presented by
SAURABH
SINGH

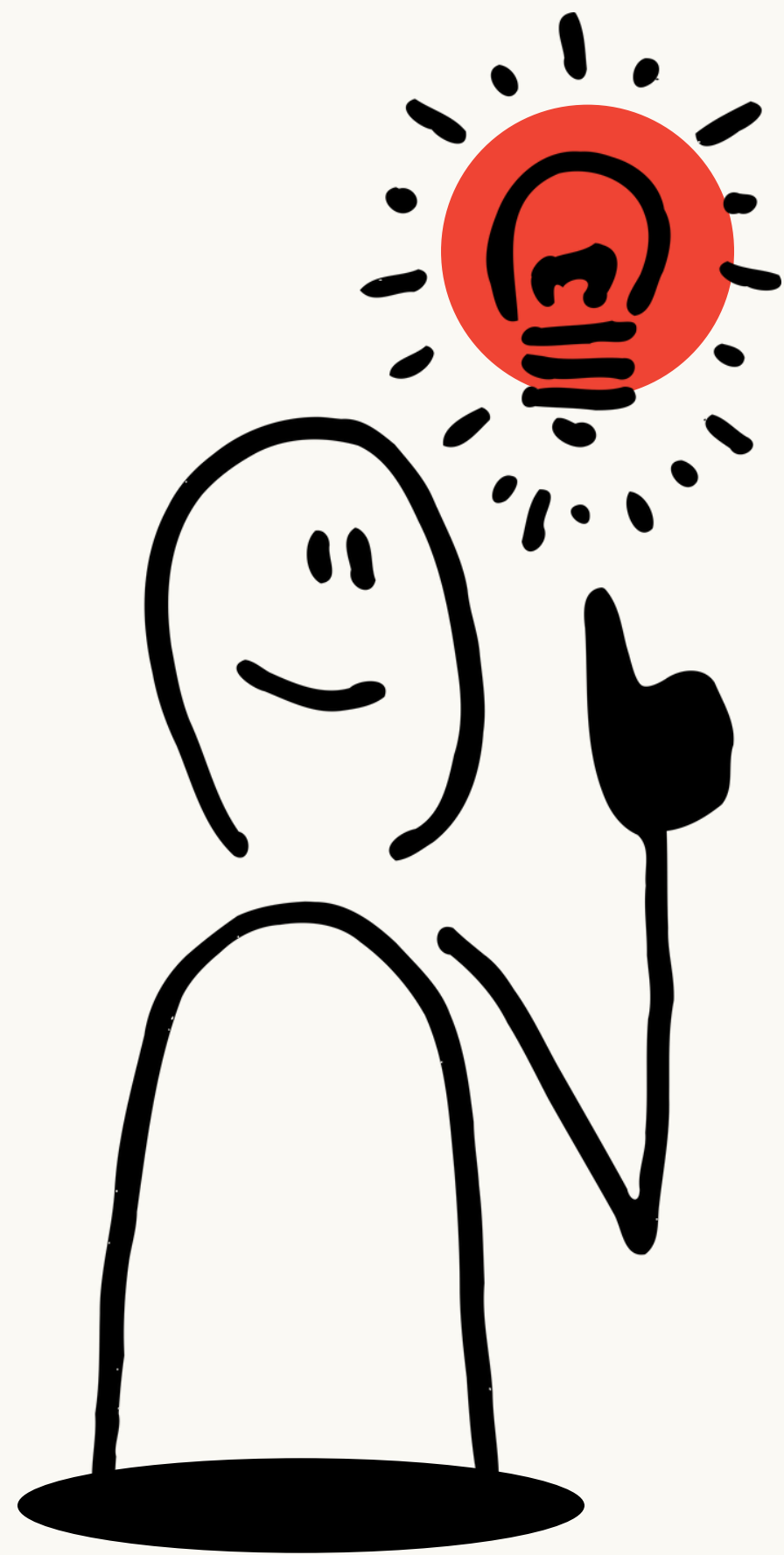


Understanding SPANCO

SQL PROJECT



How do I finally get a girlfriend?



**Wait a second... what if I apply the
SPANCO method?**

If it works in sales, maybe it works in love too!



Introduction to SPANCO

➔ SUSPECT

You don't know her, she doesn't know you—but you've seen her in the canteen thrice this week. Suspicion confirmed: she might be 'the one'... or just likes samosas.

➔ Prospect

You find out she's single. She laughed at your joke once. There's potential. It's time to run a SELECT query on your chances.

➔ Approach

You finally say 'Hi'. She says 'Oh hey, you're in IT, right?' Congratulations—you've entered the friendzone schema. Time to ALTER TABLE

➔ Negotiate

She's open to coffee, but only if her friend tags along. You agree, but add a WHERE clause: 'No third wheels after date two.'

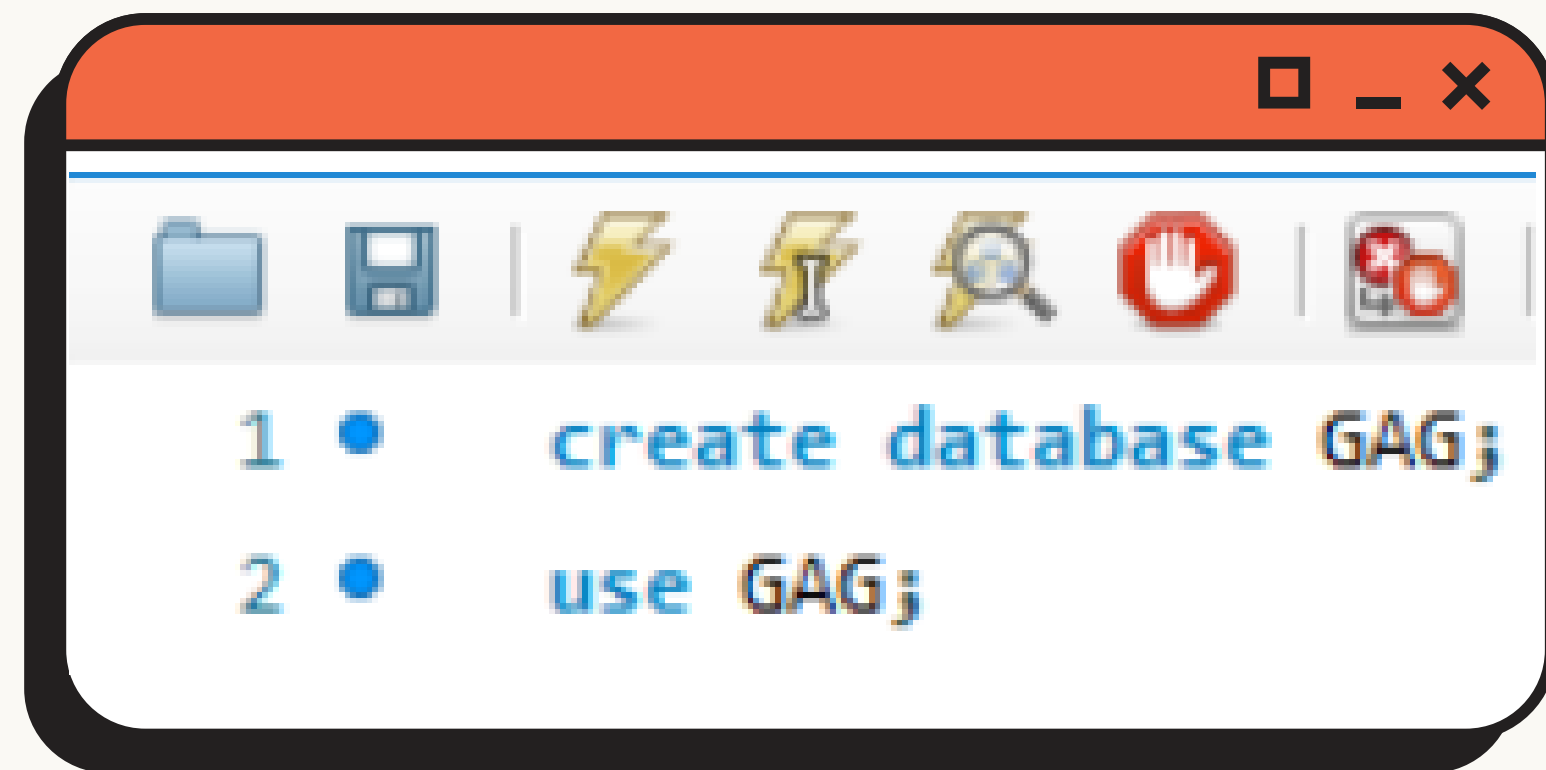
➔ Close

You both laugh at the same memes, and she doesn't ghost you. She says yes to dinner. Deal almost sealed—COMMIT pending.

➔ Order

You're officially dating. Status = 'In Relationship'. You've INSERTED INTO love_table successfully.

Time to Code: Creating the GAG Database 🍷



Love is chaos. SQL brings structure.

```
4 • CREATE TABLE Crushes (  
5     crush_id INT AUTO_INCREMENT PRIMARY KEY,  
6     name VARCHAR(100),  
7     crush_stage ENUM('Suspect', 'Prospect', 'Approach', 'Negotiation', 'Closure', 'Order') NOT NULL,  
8     stage_updated_on DATE  
9 );
```

It all begins here—every crush logged, every stage defined. We've built a table to store the emotional rollercoaster: from silent admiration (Suspect) to dating dreams (Order). Each entry tracks who they are, where we stand, and when we made our move.



LOGGING LOVE AT FIRST SIGHT

Hmmm...never knew I would
end up playing with pawns to
get my queen

```
10 • INSERT INTO Crushes (name, crush_stage, stage_updated_on)
11 VALUES
12 ('Hrishita', 'Suspect', '2025-05-01'),
13 ('Tanya', 'Suspect', '2025-05-01'),
14 ('Meher', 'Suspect', '2025-05-01'),
15 ('Sanya', 'Suspect', '2025-05-01'),
16 ('Rhea', 'Suspect', '2025-05-01'),
17 ('Pooja', 'Suspect', '2025-05-01');
18 #suspect table
19 • select * from Crushes;
```



TANYA'S LOVE GRAPH – A FALSE ALARM

```
23 #Tanya goes till Prospect → Rejected
24 • UPDATE Crushes SET crush_stage = 'Prospect', stage_updated_on = '2025-05-02' WHERE name = 'Tanya';
25 -- She didn't vibe. Goes back to suspect
26 • UPDATE Crushes SET crush_stage = 'Suspect', stage_updated_on = '2025-05-04' WHERE name = 'Tanya';
```

Tanya was promoted to Prospect—a promising signal, maybe a vibe, maybe a sign. But alas, the stars didn't align. The conversation fizzled out, the excitement faded, and she gracefully slid back into the Suspect zone.

MEHER – THE GHOST IN THE DATABASE

```
28 #Meher goes till Approach → Ghosted
29 • UPDATE Crushes SET crush_stage = 'Prospect', stage_updated_on = '2025-05-02' WHERE name = 'Meher';
30 • UPDATE Crushes SET crush_stage = 'Approach', stage_updated_on = '2025-05-05' WHERE name = 'Meher';
31 -- Ghost ho gayi :(
32 • UPDATE Crushes SET crush_stage = 'Suspect', stage_updated_on = '2025-05-07' WHERE name = 'Meher';
```

Meher progressed with promise—Prospect → Approach. The vibes were real, the chats deep, the hope soaring. But then... poof! No replies, no closure. Just silence. And so, she was demoted back to Suspect



SANYA – TALKS GOT REAL... AND THEN FELL APART

```
34 #Sanya goes till Negotiation → Unrealistic demands
35 • UPDATE Crushes SET crush_stage = 'Prospect', stage_updated_on = '2025-05-02' WHERE name = 'Sanya';
36 • UPDATE Crushes SET crush_stage = 'Approach', stage_updated_on = '2025-05-05' WHERE name = 'Sanya';
37 • UPDATE Crushes SET crush_stage = 'Negotiation', stage_updated_on = '2025-05-07' WHERE name = 'Sanya';
38 -- Negotiation broke down
39 • UPDATE Crushes SET crush_stage = 'Suspect', stage_updated_on = '2025-05-09' WHERE name = 'Sanya';
```

Sanya made significant progress—Prospect → Approach → Negotiation. Things seemed promising until unrealistic demands entered the chat. The vibes couldn't justify the terms, and so... back to Suspect she went.

RHEA – ALMOST THERE... BUT THEN SHE GHOSTED THE GOODBYE

```
41 # Rhea goes till Closure → Changed her mind
42 • UPDATE Crushes SET crush_stage = 'Prospect', stage_updated_on = '2025-05-02' WHERE name = 'Rhea';
43 • UPDATE Crushes SET crush_stage = 'Approach', stage_updated_on = '2025-05-05' WHERE name = 'Rhea';
44 • UPDATE Crushes SET crush_stage = 'Negotiation', stage_updated_on = '2025-05-07' WHERE name = 'Rhea';
45 • UPDATE Crushes SET crush_stage = 'Closure', stage_updated_on = '2025-05-09' WHERE name = 'Rhea';
46 -- Last moment ditch
47 • UPDATE Crushes SET crush_stage = 'Suspect', stage_updated_on = '2025-05-11' WHERE name = 'Rhea';
```

Prospect → Approach → Negotiation → Closure – and then... At the final moment, she changed her mind. No warning, no drama. Just a quiet exit back to Suspect.



POOJA – EVERYTHING WAS ON TRACK.. UNTIL THE PAST CAME KNOCKING

```
49 #Pooja goes till Negotiation → Got back with ex
50 • UPDATE Crushes SET crush_stage = 'Prospect', stage_updated_on = '2025-05-02' WHERE name = 'Pooja';
51 • UPDATE Crushes SET crush_stage = 'Approach', stage_updated_on = '2025-05-05' WHERE name = 'Pooja';
52 • UPDATE Crushes SET crush_stage = 'Negotiation', stage_updated_on = '2025-05-07' WHERE name = 'Pooja';
53 -- Ex ka comeback 😞
54 • UPDATE Crushes SET crush_stage = 'Suspect', stage_updated_on = '2025-05-10' WHERE name = 'Pooja';
```

System ne 'INSERT' socha tha, par Pooja ne 'REPLACE' chala diya.

Just when things were getting serious, Prospect → Approach → Negotiation, Pooja chose a plot twist: "My ex texted me." All progress deleted. SQL updated her back to Suspect.



HRISHITA – THE ONE WHO DIDN'T GHOST!

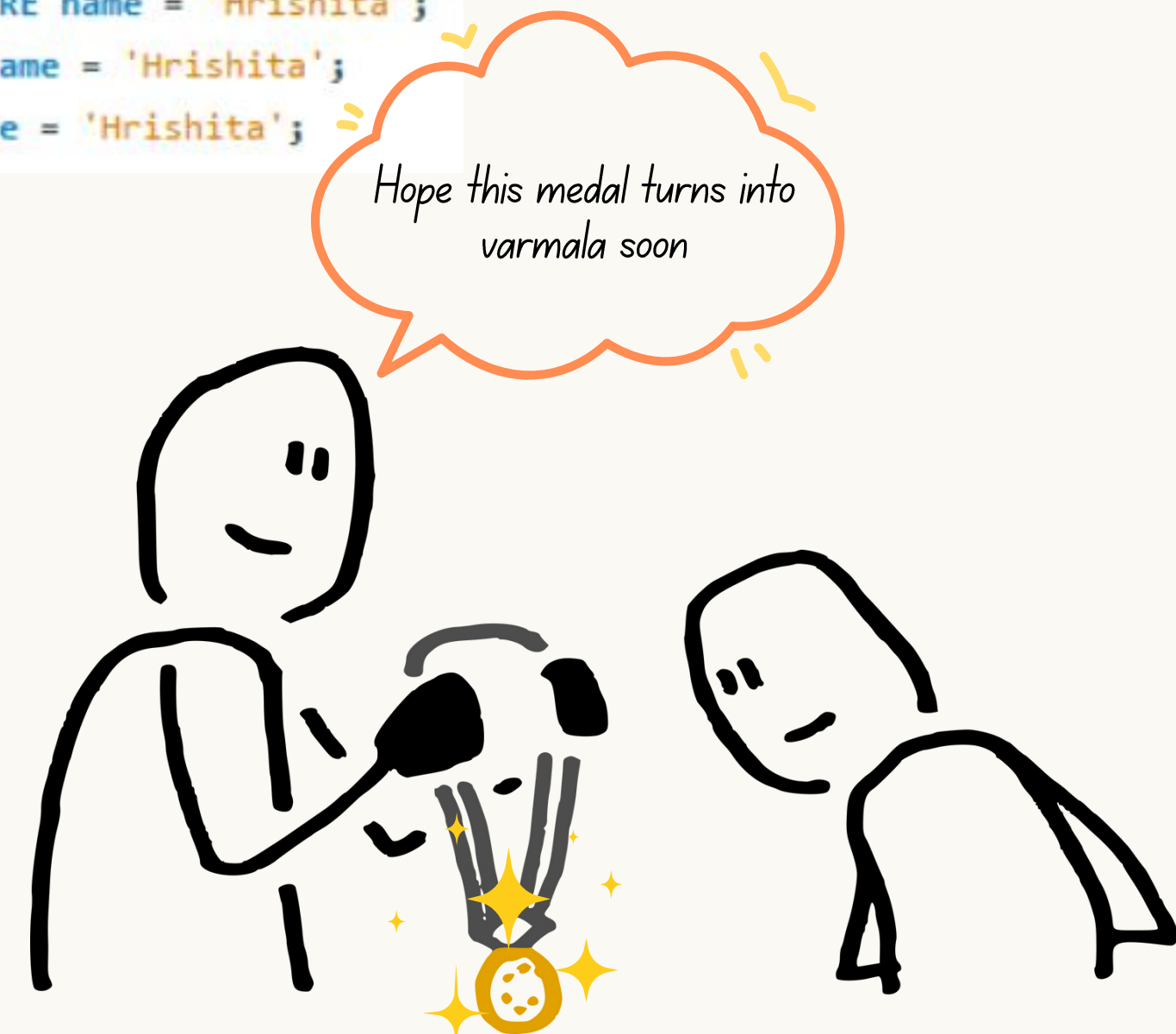
```
56 #Hrishita Reaches "Order" &
57 • UPDATE Crashes SET crush_stage = 'Prospect', stage_updated_on = '2025-05-02' WHERE name = 'Hrishita';
58 • UPDATE Crashes SET crush_stage = 'Approach', stage_updated_on = '2025-05-04' WHERE name = 'Hrishita';
59 • UPDATE Crashes SET crush_stage = 'Negotiation', stage_updated_on = '2025-05-06' WHERE name = 'Hrishita';
60 • UPDATE Crashes SET crush_stage = 'Closure', stage_updated_on = '2025-05-08' WHERE name = 'Hrishita';
61 • UPDATE Crashes SET crush_stage = 'Order', stage_updated_on = '2025-05-10' WHERE name = 'Hrishita';
```

She walked straight through every stage like a boss:

Prospect → Approach → Negotiation → Closure → Order

She didn't just UPDATE the stage...

She upgraded the whole damn system.



Crush Tracker – Final Report

```
63 #stage table
64 • SELECT name, crush_stage, stage_updated_on
65 FROM Crushes
66 ORDER BY FIELD(crush_stage, 'Suspect', 'Prospect', 'Approach', 'Negotiation', 'Closure', 'Order');
```

💡 Why FIELD()? By default, SQL orders alphabetically. **FIELD()** lets us define a custom order so that the emotional journey is tracked in the right sequence.

	name	crush_stage	stage_updated_on
	Tanya	Suspect	2025-05-04
	Meher	Suspect	2025-05-07
	Sanya	Suspect	2025-05-09
	Rhea	Suspect	2025-05-11
	Pooja	Suspect	2025-05-10
▶	Hrishita	Order	2025-05-10

