# 22AIE303 – PROJECT PHASE 3

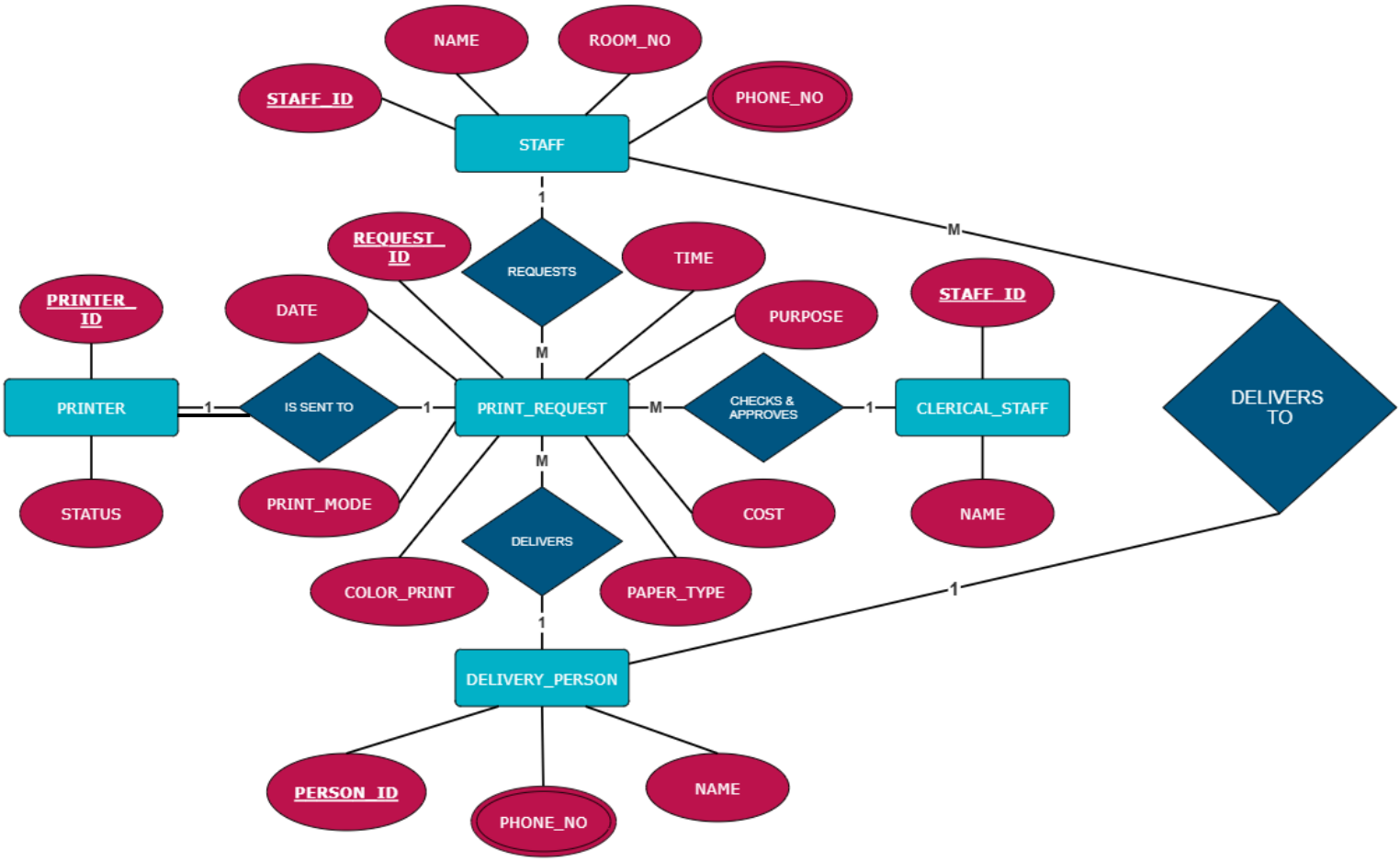
# NORMALIZATION

# GROUP 5

| Name | Roll number |
|---|---|
| Aparna Balaji | AM.EN.U4AIE22005 |
| Anuvind M P | AM.EN.U4AIE22010 |
| R S Harish Kumar | AM.EN.U4AIE22042 |
| Siddharth Menon | AM.EN.U4AIE22048 |

---

## ER-DIAGRAM :



**Original Relation** : staff_id, staff_name, staff_room_no, staff_phone (multivalued), printer_id, printer_status, clerical_staff_id, clerk_name, delivery_person_id, delivery_person_phone (multivalued), delivery_person_name, print_request_id, purpose, time, date, print-mode, color-print, paper-type, cost

## Step 1: 1NF (First Normal Form)

- In 1NF, a relation must not contain multi-valued attributes. All attributes must have atomic values.
- In the original relation, staff_phone and delivery_person_phone are multi-valued attributes.

⇨ **To transform to 1NF:**

- We need to make staff_phone and delivery_person_phone atomic (i.e., no multiple values for a single attribute in a tuple).

⇨ **Transformation:**

- **staff_phone**: Each staff member can have multiple phones, so create a new relation for staff_phone.
- **delivery_person_phone**: Similarly, create a new relation for delivery_person_phone.

⇨ **The relations in 1NF are:**

1. **Staff**(staff_id, staff_name, staff_room_no)
    - This stores information about staff members.
2. **Staff_Phone**(staff_id, staff_phone)
    - Since a staff member can have multiple phone numbers, we separate this into a new relation.
3. **Printer**(printer_id, printer_status)
    - This stores information about printers.
4. **Clerical_Staff**(clerical_staff_id, clerk_name)
    - Information about clerical staff.
5. **Delivery_Person**(delivery_person_id, delivery_person_name)
    - Information about delivery persons.
6. **Delivery_Phone**(delivery_person_id, delivery_person_phone)
    - Since a delivery person can have multiple phone numbers, we create a separate relation for this.
7. **Print_Request**(print_request_id, staff_id, printer_id, purpose, time, date, print_mode, color_print, paper_type, cost)
    - Stores information about print requests made by staff, using a printer, with various print settings and associated costs.

⇨ **Final 1NF Relations:**

1. **Staff**(staff_id, staff_name, staff_room_no)
2. **Staff_Phone**(staff_id, staff_phone)
3. **Printer**(printer_id, printer_status)
4. **Clerical_Staff**(clerical_staff_id, clerk_name)
5. **Delivery_Person**(delivery_person_id, delivery_person_name)
6. **Delivery_Phone**(delivery_person_id, delivery_person_phone)
7. **Print_Request**(print_request_id, staff_id, printer_id, purpose, time, date, print_mode, color_print, paper_type, cost)

---

## Step 2: 2NF (Second Normal Form)

2NF Requirements:

- The relation must be in 1NF.
- All non-prime attributes must be fully functionally dependent on the whole primary key. In other words, we need to eliminate partial dependencies, where a non-prime attribute depends only on part of a composite primary key.

**Analysis of Partial Dependencies:**

- **Print_Request** has a **composite key** (print_request_id, staff_id, printer_id).
- The **partial dependency** occurs because staff_name and staff_room_no depend only on staff_id and not on the whole composite key of Print_Request. Similarly, printer_status depends only on printer_id.

**Decomposition for 2NF:**

- Create a new relation for staff attributes (staff_name, staff_room_no) that depend only on staff_id.
- Create a new relation for printer attributes (printer_status) that depend only on printer_id

**New Relations in 2NF:**

1. **Staff**(staff_id, staff_name, staff_room_no) — *No partial dependency, staff_name and staff_room_no depend fully on staff_id.*

2. **Staff_Phone**(staff_id, staff_phone) — *No partial dependency, each staff_id can have multiple staff_phone numbers.*
3. **Printer**(printer_id, printer_status) — *No partial dependency, printer_status depends fully on printer_id.*
4. **Clerical_Staff**(clerical_staff_id, clerk_name) — *This relation remains unchanged as clerk_name depends fully on clerical_staff_id.*
5. **Delivery_Person**(delivery_person_id, delivery_person_name) — *No partial dependency as delivery_person_name depends fully on delivery_person_id.*
6. **Delivery_Phone**(delivery_person_id, delivery_person_phone)
7. **Print_Request**(print_request_id, staff_id, printer_id, purpose, time, date, print_mode, color_print, paper_type, cost) — *Now this only contains attributes that depend on the whole composite key (print_request_id, staff_id, printer_id).*

---

## Step 3: Convert to 3NF (Third Normal Form)

**3NF Requirements:**

- The relation must be in **2NF**.
- There should be **no transitive dependencies**, i.e., non-prime attributes must not depend on other non-prime attributes.

**Analysis of Transitive Dependencies:**

- In **Print_Request**, the **attribute cost** might depend on printer_id (since the cost can vary based on the printer used), and printer_id is part of the composite key. This creates a **transitive dependency**: cost depends on printer_id, which is part of the primary key, but cost indirectly depends on the print_request_id through printer_id.

We need to remove this transitive dependency by creating a new relation for printer_id and cost.

**Decomposition for 3NF:**

- Create a new relation Printer_Cost that contains printer_id and cost.

**New Relations in 3NF:**

1. **Staff**(staff_id, staff_name, staff_room_no) — No transitive dependency.
2. **Staff_Phone**(staff_id, staff_phone) — No transitive dependency.
3. **Printer**(printer_id, printer_status) — No transitive dependency.

4. **Clerical_Staff**(clerical_staff_id, clerk_name) — No transitive dependency.
5. **Delivery_Person**(delivery_person_id, delivery_person_name) — No transitive dependency.
6. **Delivery_Phone**(delivery_person_id, delivery_person_phone) — No transitive dependency.
7. **Print_Request**(print_request_id, staff_id, printer_id, purpose, time, date, print_mode, color_print, paper_type) — No transitive dependency for cost.
8. **Printer_Cost**(printer_id, cost) — The cost attribute is now fully dependent on printer_id.

**Final Relations in 3NF:**

1. **Staff**(staff_id, staff_name, staff_room_no)
2. **Staff_Phone**(staff_id, staff_phone)
3. **Printer**(printer_id, printer_status)
4. **Clerical_Staff**(clerical_staff_id, clerk_name)
5. **Delivery_Person**(delivery_person_id, delivery_person_name)
6. **Delivery_Phone**(delivery_person_id, delivery_person_phone)
7. **Print_Request**(print_request_id, staff_id, printer_id, purpose, time, date, print_mode, color_print, paper_type)
8. **Printer_Cost**(printer_id, cost)