Step 8.1 Create a value-independent view that hides some private information.

```
CREATE VIEW Public_Ward AS
SELECT
   Ward_Number,
   Ward_Name,
   Ward_Location
FROM
   Ward;

SELECT * FROM Public_Ward;
---------------
CREATE VIEW Public_Patient AS
SELECT
 Patient_Number,
 Patient_Name,
 Address
FROM
 Patient;

SELECT * FROM Public_Patient;

---------------

CREATE VIEW Public_Medication AS
SELECT
   Medication_Name,
   Patient_Number,
   Drug_Number
FROM
   Medication;

SELECT * FROM Public_Medication;

---------------

CREATE VIEW Public_Staff AS
SELECT
 Staff_Number,
```

```
  Staff_Name,
  Qualification
FROM
  Staff;

SELECT * FROM Public_Staff;

---------------

CREATE VIEW Public_Supply AS
SELECT
    Supply_Number,
    Supply_Name,
    Quantity_In_Stock
FROM
    Supply;

SELECT * FROM Public_Supply;

---------------

CREATE VIEW Public_Requisition AS
SELECT
  Requisition_Number,
  Quantity_Required,
  Date_Of_Order,
  Delivery_Date
FROM
  Requisition;

SELECT * FROM Public_Requisition;

---------------

CREATE VIEW Public_Supplier AS
SELECT
  Supplier_Number,
  Supplier_Name,
```

```
 Address
FROM
 Supplier;


SELECT * FROM Public_Supplier;
```

Step 8.2 Create a value-dependent view that screens some data.

```
CREATE VIEW Patient_Age AS
SELECT Patient_Number, Patient_Name, Age, Address, Contact_Number, Ward_Number
FROM Patient
WHERE Age > 60;


SELECT * FROM Patient_Age;


--------------


CREATE VIEW Staff_Past_Experience AS
SELECT Staff_Number, Staff_Name, Qualification, Past_Experience, Ward_Number
FROM Staff
WHERE Staff_Name = "Isabella Turner" AND Ward_Number = 12;


SELECT * FROM Staff_Past_Experience;


--------------


CREATE VIEW Supplier_FullName AS
SELECT Supplier_Number, Supplier_Name, Address, Email, Telephone_Number, Fax_Number
FROM Supplier
WHERE Supplier_Name LIKE 'a%';


SELECT * FROM Supplier_FullName;
```

Step 8.3 Set up an audit trail for updates to a sensitive item that users can update and test it by updating the item.

```
CREATE TABLE IF NOT EXISTS AuditTrail (
  AuditTrail_ID INT AUTO_INCREMENT PRIMARY KEY,
  TableName VARCHAR(100) NOT NULL,
  Action VARCHAR(10) NOT NULL,
  Old_Values JSON,
  New_Values JSON,
  Update_DateTime TIMESTAMP DEFAULT CURRENT_TIMESTAMP
) ENGINE=InnoDB;



DELIMITER //
CREATE TRIGGER trg_Patient_AuditTrail_Update
AFTER UPDATE ON Patient FOR EACH ROW
BEGIN
  INSERT INTO AuditTrail (TableName, Action, Old_Values, New_Values)
  VALUES ('Patient', 'UPDATE', JSON_OBJECT(
    'Patient_Number', OLD.Patient_Number,
    'Patient_Name', OLD.Patient_Name,
    'Age', OLD.Age,
    'Address', OLD.Address,
    'Contact_Number', OLD.Contact_Number,
    'Ward_Number', OLD.Ward_Number
  ), JSON_OBJECT(
    'Patient_Number', NEW.Patient_Number,
    'Patient_Name', NEW.Patient_Name,
    'Age', NEW.Age,
    'Address', NEW.Address,
    'Contact_Number', NEW.Contact_Number,
    'Ward_Number', NEW.Ward_Number
  ));
END;
//
DELIMITER;
```

```
UPDATE Patient SET Age = 36 WHERE Patient_Number = 1;


SELECT * FROM AuditTrail;
```