

1. BEFORE INSERT Trigger:

```
CREATE OR REPLACE TRIGGER before_insert_trigger
BEFORE INSERT ON employees
FOR EACH ROW
BEGIN
    -- Set current date as the hire date if it is not provided
    IF :NEW.hire_date IS NULL THEN
        :NEW.hire_date := SYSDATE;
    END IF;
END;
/
```

2. AFTER INSERT Trigger:

```
CREATE OR REPLACE TRIGGER after_insert_trigger
AFTER INSERT ON employees
FOR EACH ROW
BEGIN
    -- Log the inserted employee information into a separate audit table
    INSERT INTO audit_table (employee_id, action, action_date)
    VALUES (:NEW.employee_id, 'INSERT', SYSDATE);
END;
/
```

3. BEFORE UPDATE Trigger:

```
CREATE OR REPLACE TRIGGER before_update_trigger
BEFORE UPDATE ON employees
FOR EACH ROW
BEGIN
    -- Prevent updating the salary to a lower value
    IF :NEW.salary < :OLD.salary THEN
        raise_application_error(-20001, 'Salary cannot be decreased.');
```

4. AFTER UPDATE Trigger:

```
CREATE OR REPLACE TRIGGER after_update_trigger
AFTER UPDATE ON employees
FOR EACH ROW
BEGIN
    -- Log the updated employee information into a separate audit table
```

```
INSERT INTO audit_table (employee_id, action, action_date)
VALUES (:OLD.employee_id, 'UPDATE', SYSDATE);
END;
/
```

5. BEFORE DELETE Trigger:

```
CREATE OR REPLACE TRIGGER before_delete_trigger
BEFORE DELETE ON employees
FOR EACH ROW
BEGIN
    -- Prevent deleting employees with a higher salary than a threshold
    IF :OLD.salary > 10000 THEN
        raise_application_error(-20001, 'Employees with salary higher than 10000
cannot be deleted.');
```

```
END IF;
```

```
END;
/
```

6. AFTER DELETE Trigger:

```
CREATE OR REPLACE TRIGGER after_delete_trigger
AFTER DELETE ON employees
FOR EACH ROW
BEGIN
    -- Log the deleted employee information into a separate audit table
    INSERT INTO audit_table (employee_id, action, action_date)
    VALUES (:OLD.employee_id, 'DELETE', SYSDATE);
END;
/
```

In each trigger, the table name is set to "employees" for illustration purposes. Replace it with the actual name of your table. Adjust the trigger logic within the **BEGIN** and **END** block as per your specific requirements for each trigger type.