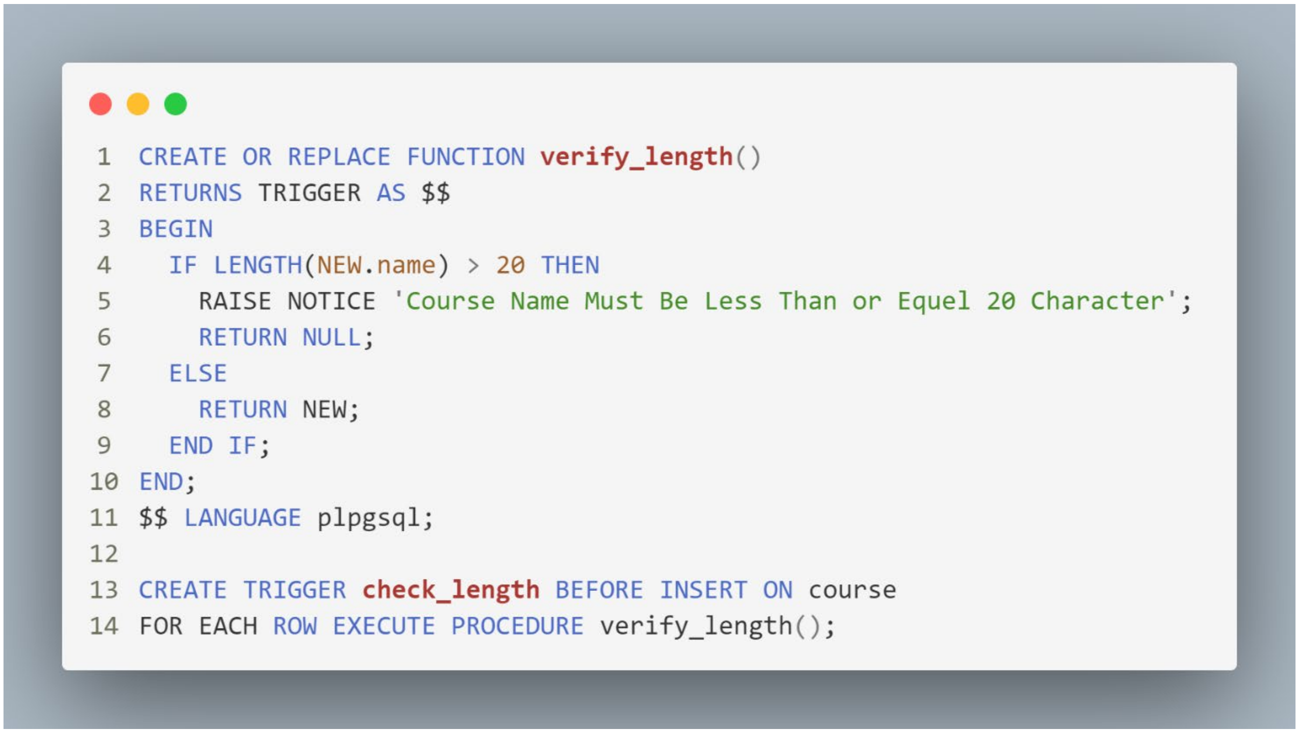
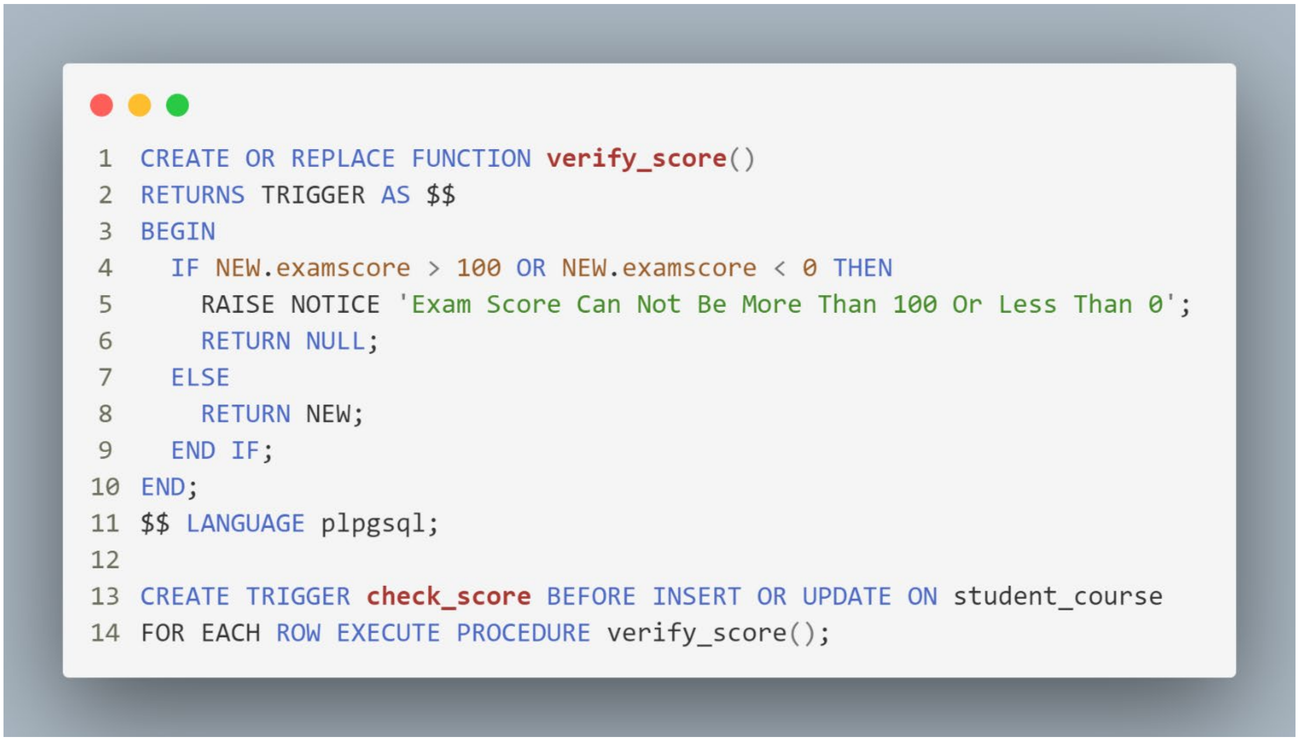
1. Create trigger to prevent insert new Course with name length greater than 20 chars;



```sql
CREATE OR REPLACE FUNCTION verify_length()
RETURNS TRIGGER AS $$
BEGIN
  IF LENGTH(NEW.name) > 20 THEN
    RAISE NOTICE 'Course Name Must Be Less Than or Equel 20 Character';
    RETURN NULL;
  ELSE
    RETURN NEW;
  END IF;
END;
$$ LANGUAGE plpgsql;

CREATE TRIGGER check_length BEFORE INSERT ON course
FOR EACH ROW EXECUTE PROCEDURE verify_length();
```

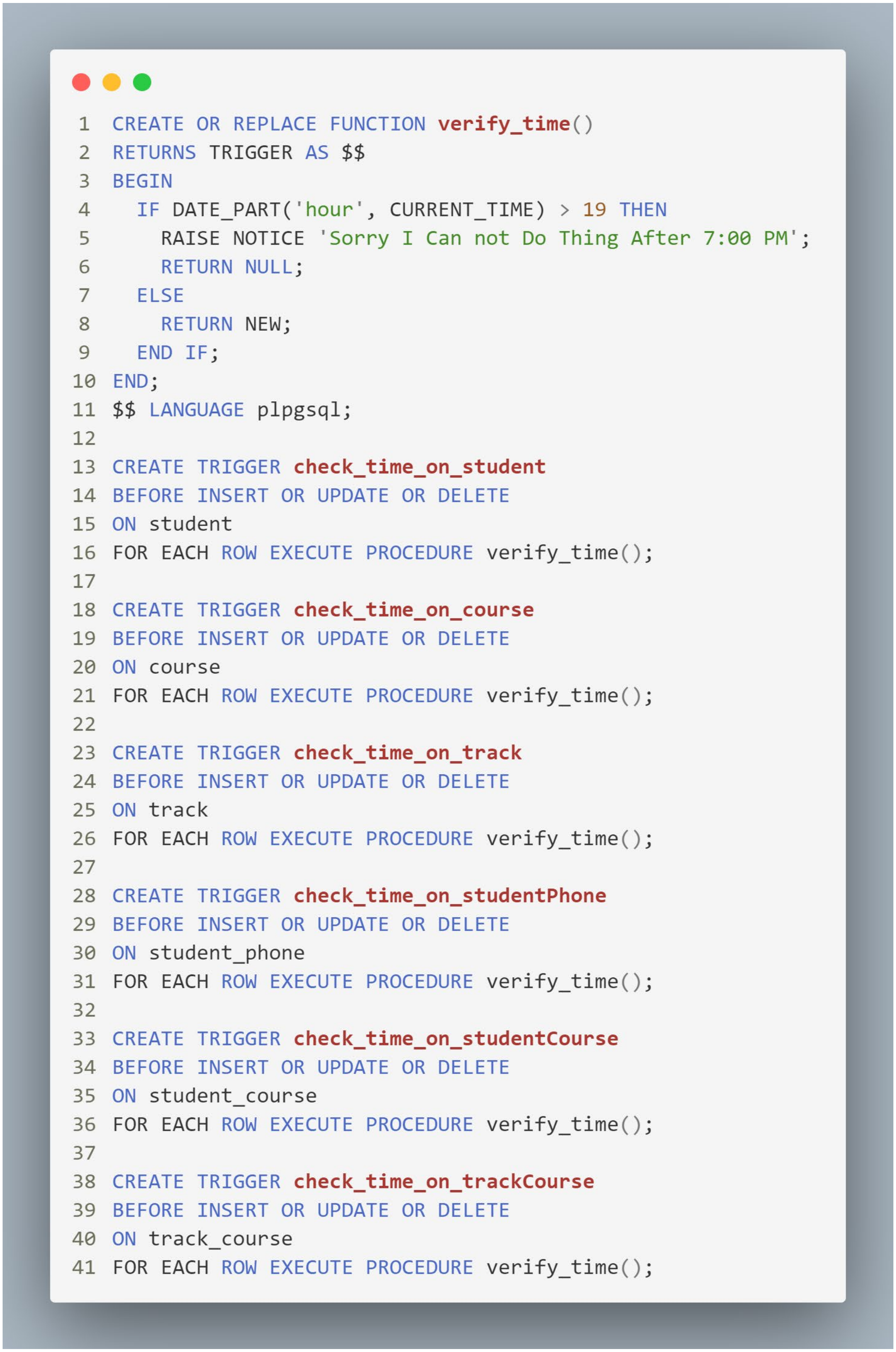
2. Create trigger to prevent user to insert or update Exam with Score greater than 100 or less than zero



```sql
CREATE OR REPLACE FUNCTION verify_score()
RETURNS TRIGGER AS $$
BEGIN
  IF NEW.examscore > 100 OR NEW.examscore < 0 THEN
    RAISE NOTICE 'Exam Score Can Not Be More Than 100 Or Less Than 0';
    RETURN NULL;
  ELSE
    RETURN NEW;
  END IF;
END;
$$ LANGUAGE plpgsql;

CREATE TRIGGER check_score BEFORE INSERT OR UPDATE ON student_course
FOR EACH ROW EXECUTE PROCEDURE verify_score();
```

3. (bonus) Create trigger to prevent any user to update/insert/delete to all tables (Students, Exams, Tracks,..) after 7:00 PM



```
CREATE OR REPLACE FUNCTION verify_time()
RETURNS TRIGGER AS $$
BEGIN
  IF DATE_PART('hour', CURRENT_TIME) > 19 THEN
    RAISE NOTICE 'Sorry I Can not Do Thing After 7:00 PM';
    RETURN NULL;
  ELSE
    RETURN NEW;
  END IF;
END;
$$ LANGUAGE plpgsql;

CREATE TRIGGER check_time_on_student
BEFORE INSERT OR UPDATE OR DELETE
ON student
FOR EACH ROW EXECUTE PROCEDURE verify_time();

CREATE TRIGGER check_time_on_course
BEFORE INSERT OR UPDATE OR DELETE
ON course
FOR EACH ROW EXECUTE PROCEDURE verify_time();

CREATE TRIGGER check_time_on_track
BEFORE INSERT OR UPDATE OR DELETE
ON track
FOR EACH ROW EXECUTE PROCEDURE verify_time();

CREATE TRIGGER check_time_on_studentPhone
BEFORE INSERT OR UPDATE OR DELETE
ON student_phone
FOR EACH ROW EXECUTE PROCEDURE verify_time();

CREATE TRIGGER check_time_on_studentCourse
BEFORE INSERT OR UPDATE OR DELETE
ON student_course
FOR EACH ROW EXECUTE PROCEDURE verify_time();

CREATE TRIGGER check_time_on_trackCourse
BEFORE INSERT OR UPDATE OR DELETE
ON track_course
FOR EACH ROW EXECUTE PROCEDURE verify_time();
```

4. Backup your Database to external file
    - pg_dump iti > backup.sql;
5. Backup your Student table to external file
    - COPY student TO 'D:\Web\ITI\ITI_Tasks\PostgreSQL\student.sql';