

```
-- Triggers de restrição
SET SCHEMA 'urblixo';
```

```
-- Uma trigger que garante que um motorista dirige um e apenas um caminhao
```

```
CREATE OR REPLACE FUNCTION verifica_motorista_caminhao()
```

```
RETURNS TRIGGER AS $$
```

```
DECLARE
```

```
    qtd_motoristas INTEGER;
```

```
BEGIN
```

```
    SELECT COUNT(*) INTO qtd_motoristas
```

```
    FROM motorista
```

```
    WHERE id_caminhao = NEW.id_caminhao;
```

```
    IF qtd_motoristas > 0 THEN
```

```
        RAISE EXCEPTION 'Este caminhão já possui um motorista associado.';
```

```
    END IF;
```

```
    RETURN NEW;
```

```
END;
```

```
$$ LANGUAGE plpgsql;
```

```
-- Associar a trigger à tabela motorista
```

```
CREATE TRIGGER trigger_verificar_motorista_caminhao
```

```
BEFORE INSERT ON motorista
```

```
FOR EACH ROW
```

```
EXECUTE FUNCTION verifica_motorista_caminhao();
```

```
-- Teste da trigger que verificar a associação de motorista com caminhao
```

```
INSERT INTO motorista (cpf, nome, id_caminhao) VALUES ('12345678910', 'Fulano', 2);
```

```
-- Uma trigger que não permite que um ponto seja excluído quando ele tem uma rota associada a ele
```

```
CREATE OR REPLACE FUNCTION evitar_exclusao_pontos_referenciados()
```

```
RETURNS TRIGGER AS $$
```

```
BEGIN
```

```
    IF EXISTS (SELECT 1 FROM rota_de_coleta WHERE id_ponto = OLD.id_ponto) THEN
```

```
        RAISE EXCEPTION 'Não é possível excluir este ponto de coleta pois ele está sendo referenciado p  
or rotas de coleta.';
```

```
    END IF;
```

```
    RETURN OLD;
```

```
END;
```

```
$$ LANGUAGE plpgsql;
```

```
CREATE TRIGGER trg_evitar_exclusao_pontos_referenciados
```

```
BEFORE DELETE ON ponto_de_coleta
```

```
FOR EACH ROW
```

```
EXECUTE FUNCTION evitar_exclusao_pontos_referenciados();
```

```
-- Tente excluir o ponto de coleta
```

```
DELETE FROM ponto_de_coleta WHERE id_ponto = 1;
```