

Projeto de Bases de Dados

Parte 2

Eduardo Janicas: 78974 (4H)

Diana Antunes: 82448 (6H)

Nuno Fernande: 80774 (4H)

Número de grupo: 44

Turno: BD225179L06

Modelo Relacional

User(NIF, telefone, nome)

Edifício(Morada)

Contém_alugável(Morada, Código, foto)

Morada: FK Edifício(Morada)

Arrenda(Morada, Código, NIF)

Morada, Código: FK Contém_alugável(Morada, Código)

NIF: FK User(NIF)

Fiscal(ID, Empresa)

Fiscaliza(ID, Morada, Código)

ID: FK Fiscal(ID)

Espaço(Morada, Código)

Morada, Código: FK Contém_alugável(Morada, Código)

Posto(morada, código, espaço_Morada, espaço_Código)

espaço_Morada, espaço_Código: FK Espaço(Morada, Código)

Oferece_oferta(Morada, Código, data_início, data_fim, Tarifa)

Morada, Código: FK Contém_alugável(Morada, Código)

Reserva(Número)

Paga(Número, data, método)

Número: FK Reserva(Número)

Estado_log(Número, Timestamp, estado)

Número: FK Reserva(Número)

Aluga(NIF, Morada, Código, data_início, Número)

NIF: FK User(NIF)

Morada, Código, data_início: FK Oferece_oferta(Morada, Código, data_início)

Número: FK Reserva(Número)

Restrições de Integridade

1. Posto AND Espaço COVERS Alugável
2. O atributo “estado” das entidades “Reserva” pode conter os valores “Pendente”, “Aceite”, “Declinada” ou “Cancelada”.
3. Uma reserva só pode ser paga se o estado atual for “Aceite”.
4. No máximo, só pode existir uma reserva aceite sobre cada oferta.
5. As ofertas para o mesmo alugável não se podem sobrepor no tempo.
6. O atributo código da entidade alugável deve ser um número sequencial.

No modelo EA, não é possível criar um posto sem um espaço válido associado, mas no Modelo Relacional a Foreign Key pode ter o valor NULL. Portanto, temos de adicionar a seguinte Restrição de Integridade:

- 7: Cada Posto, quando é criado, tem de ser associado a um Espaço válido.

Álgebra Relacional

1. Liste a morada de todos os edifícios que contêm pelo menos um alugável com mais de uma oferta

$$Ofertas \leftarrow \pi_{morada,código,data_inicio}(Contém_alugável \bowtie Oferece_oferta)$$

ParesOfertas

$$\leftarrow \rho_{1 \rightarrow morada1, 2 \rightarrow código1, 3 \rightarrow data_inicio1, 4 \rightarrow morada2, 5 \rightarrow código2, 6 \rightarrow data_inicio2} (Ofertas \times Ofertas)$$

$$\pi_{morada1}(\sigma_{(código1=código2 \wedge data_inicio1 \neq data_inicio2)}(ParesOfertas))$$

Alternativamente ao produto externo podia ser usada uma função de agregação e devolver quem tivesse um COUNT da oferta > 1.

2. Liste o estado atual de todas as reservas pagas

$$\pi_{estado}(número, estado \mathrel{\mathcal{G}}_{\max(timestamp)} Estado_log \bowtie Paga)$$

Consideramos o estado atual como o estado com maior timestamp

3. Liste o identificador completo dos espaços de trabalho cujos postos nele contidos foram todos alugados

$$Posto / (\pi_{morada,código}(\sigma_{estado="aceite"}(OfereceOferta \bowtie Reserva \bowtie Estado)))$$

SQL

1. Liste a morada de todos os edifícios que contêm pelo menos um alugável com mais de uma oferta

```
SELECT morada FROM
    Contém_alugável NATURAL JOIN Oferece_oferta
GROUP BY morada, código
HAVING COUNT(*) > 1
```

2. Liste o estado atual de todas as reservas pagas

```
SELECT estado FROM
    Estado_log NATURAL JOIN Paga
GROUP BY número, estado
HAVING MAX(timestamp)
```