

Base de dados - LIVROS - Biblioteca

Grupo 33

Turno p3

Afonso Batista Nº. 59975

Francisco Freitas Nº. 60313

Sebastião Jerónimo Nº. 60840

Aplicação desenvolvida em Oracle - Application Express - V 5.1.3.00.05

Bibliografia usada na 1 fase:

Capítulo 7 e 2.1 a 2.4 do livro recomendado.

Capítulo 2 e 3.1 a 3.3 do livro de Ramakrishnan et al.

Capítulo 1 do livro de David Maier.

slides de BD aulas teóricas

DESCRIÇÃO

Uma Biblioteca disponibiliza livros para serem requisitados.

Esses livros estão identificados pelo ID.

Os livros também têm um nome, ISBN, data de publicação, pelo menos um autor, o género literário, o número de edição e o nome da editora.

Os livros que podem ser requisitados pelas pessoas estão numa estante.

Sendo que essa estante apenas terá livros do mesmo género literário e um número único

Numa biblioteca pode haver várias salas e em cada sala podem existir várias estantes.

Cada pessoa pode requisitar vários livros.

A biblioteca tem registadas as correspondências entre os livros e os seus autores.

Para cada pessoa que requisita quer se saber algumas informações como por exemplo o nome, o seu endereço de email, NIF, data de nascimento.

O processo mais importante é o facto de um leitor poder requisitar um livro em que nesta transação a biblioteca faz o empréstimo de um livro a um leitor que passa a ser requisitado, através de um bibliotecário. O bibliotecário irá permitir a requisição dos livros que cada pessoa quer requisitar da biblioteca.

Também existe um funcionário de Limpeza que irá limpar certas salas da biblioteca.

Para cada tipo de funcionário (bibliotecário ou funcionário de Limpeza) quer saber-se o número de funcionário no estabelecimento com o NIF, carga horária semanal, ordenado, e a descrição do trabalho.

Objetivos da base de Dados:

A partir desta base de dados conseguimos ter acesso a todos os livros de uma biblioteca e as suas informações como por exemplo os seus autores ou ISBN, que estejam numa estante ou tenham sido requisitados.

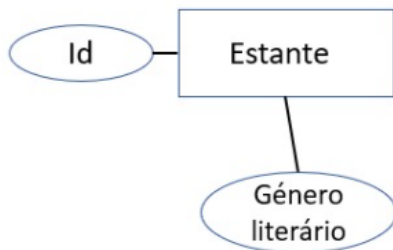
Caso estejam requisitados é possível aceder a cada empréstimo e a cada leitor e às suas informações, ao respetivo livro requisitado, quando ocorreu a requisição e ainda ao profissional bibliotecário que fez o empréstimo.

Ou saber quais salas são limpas por um certo funcionário de limpeza. Com esta base de dados também podemos ter acesso a todos os funcionários de uma biblioteca, aos seus salários, as suas cargas horárias e às descrições das suas funções/tarefas.

ENTIDADES

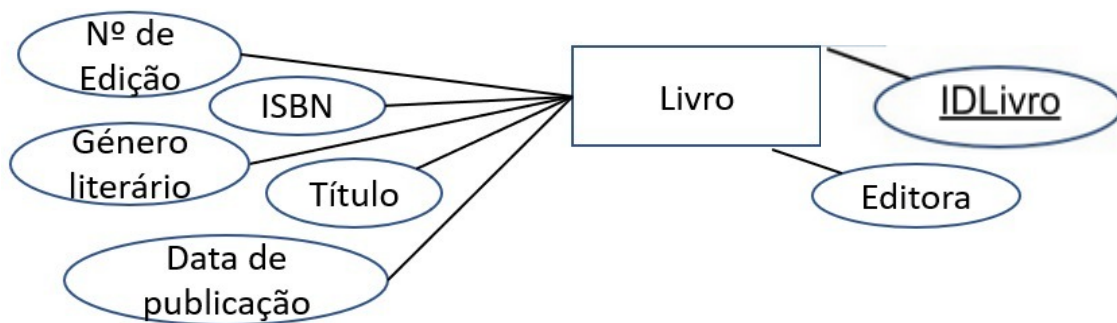
- Estante:

Os seus atributos : número único da estante (id) e o género literário.
id chave primária



-Livro :

Os seus atributos : IDLivro, título, número de edição, data de publicação, pelo menos um autor, o ISBN, género literário e o nome da editora.
IDLivro é a chave primária

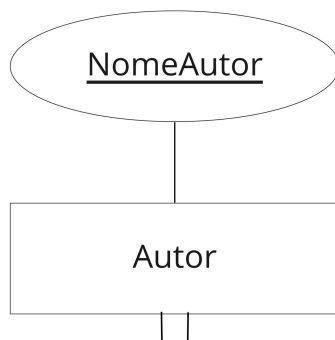


-Autor :

nomeAutor é a chave primária

Os seus atributos : o nomeAutor

O autor, neste caso, é uma entidade pois um livro pode ter N autores e é necessário expressar a sua relação de (N:M), pois como atributo não há a possibilidade de dizer que um autor pode ter mais que um livro. Além disso, evita a repetição na tabela do nome do autor, caso existam muitos livros desse autor.



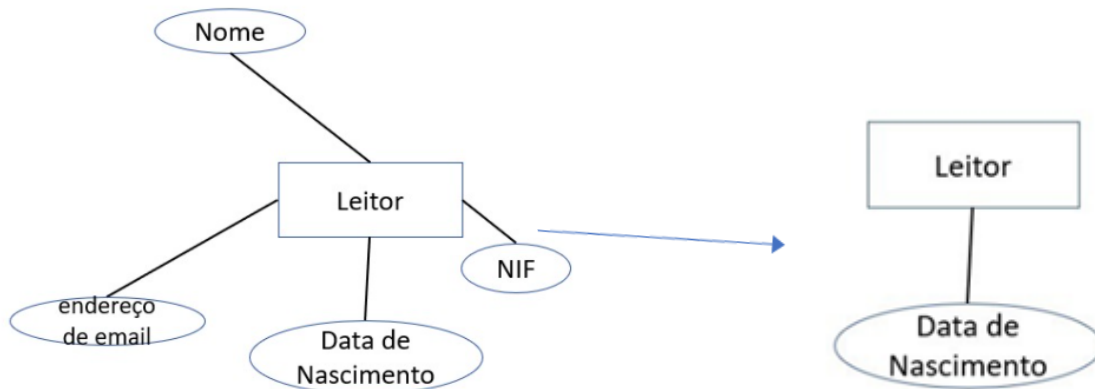
-Leitor:

Esta entidade poderá requisitar livros.

Leitor(NIF , data de nascimento)

NIF é chave estrangeira de Pessoa

O segundo modelo para a entidade leitor é com a ISA pessoa



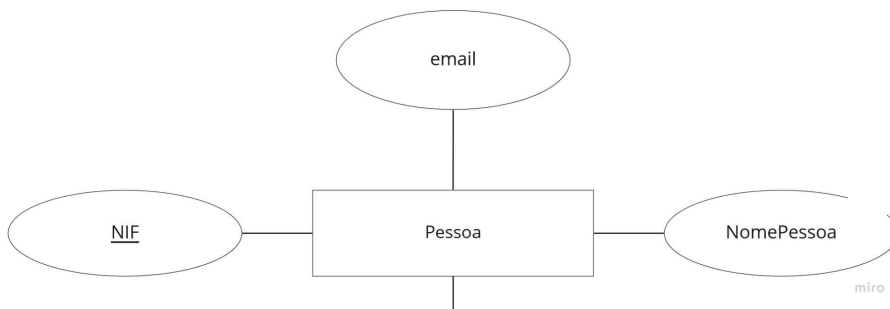
-Pessoa:

os seus atributos : NomePessoa, NIF,email

Pessoa(NIF,NomePessoa,email)

NIF é a chave primária

Chegamos a esta entidade devido ao facto de haver redundância, ou seja, repetição de atributos iguais no funcionário e no leitor.



-Funcionário:

Os seus atributos: (NIF,nFuncionário)

NIF é chave estrangeira de pessoas

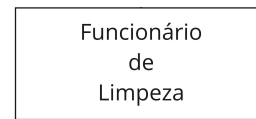
(Colocámos o atributo nFuncionario pois numa empresa não é habitual gerir os funcionários através do NIF. Para além dos pagamentos, faz sentido ter um número de funcionário porque, por exemplo, é mais fácil alguém dizer que o funcionário número 13 é responsável pelo turno da tarde do que tratá-lo pelo NIF, garantindo uma maior legibilidade.

Estes dividem-se em bibliotecários e funcionários de limpeza.

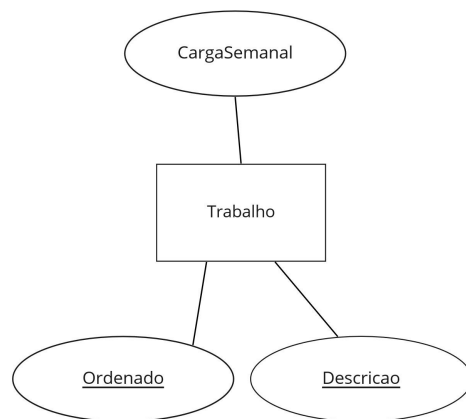


-Bibliotecário:
Bibliotecario(NIF)
NIF é a chave estrangeira de funcionário.

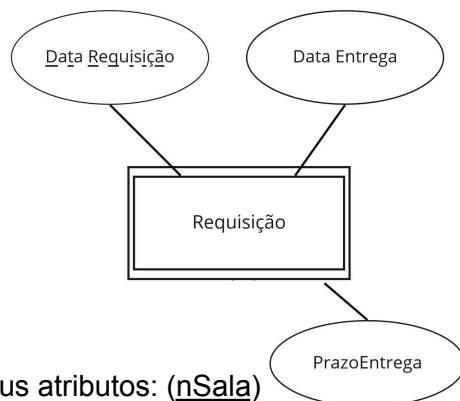
Funcionário de Limpeza
Funcionário de Limpeza(NIF)
NIF é a chave estrangeira de funcionário.



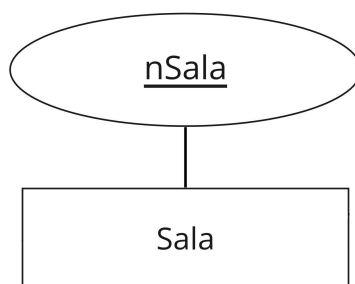
-Trabalho
Os seus atributos:(cargaSemanal, ordenado,descricao)
Ordenado e descricao são as chaves primárias



-Requisicao
Os seus atributos: (DataRequisicao,DataEntrega,PrezoEntrega)
DataRequisicao é a chave primária

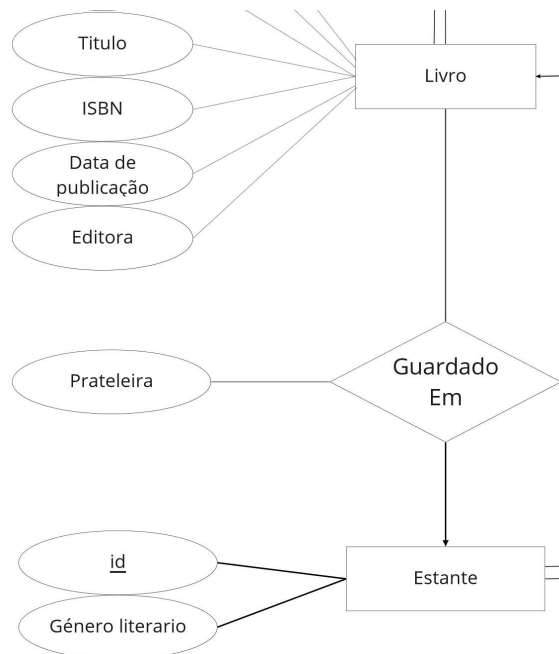


Sala
Os seus atributos: (nSala)
nSala é chave primária de Sala



RELAÇÕES

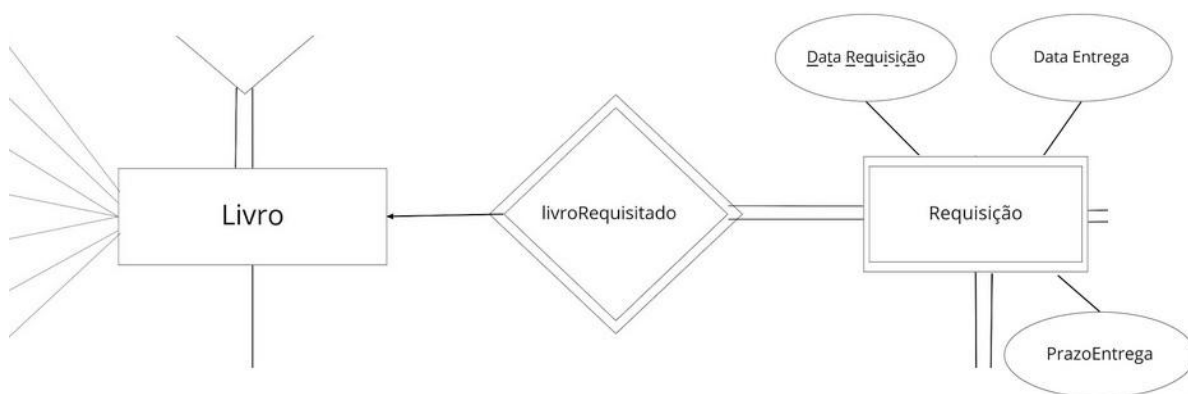
A relação “GuardadoEm” é uma relação que pretende relacionar o livro que ainda está na biblioteca com a estante e prateleira onde está (relaciona as entidades Livro e Estante). A relação é de (N:1) pois cada livro está numa só estante e é total, pois todo o livro não requisitado tem que ter uma estante. É parcial no lado do conjunto de entidades Estante, pois nem todas as estantes podem ter livros.



A relação “Livro requisitado”
Esta relação fraca
várias ocorrências
requisitado várias
diferentes pelo

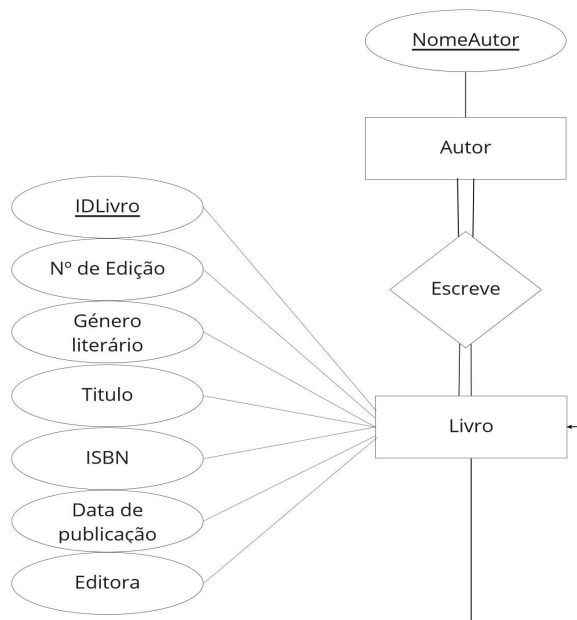
Requisitado” serve para ligar
à entidade fraca Requisicao.
serve para diferenciar as
de o mesmo livro ter sido
vezes em momentos
mesmo Leitor.

A relação é total no lado da entidade Requisicao pois qualquer requisicao tem apenas um livro associado, mas nem todo o livro terá uma requisicao associada. Então será uma relação (N:1).



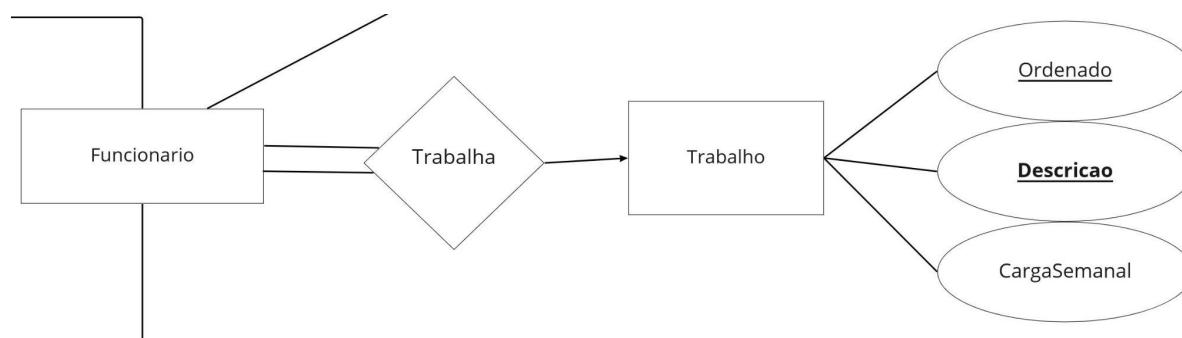
A relação “Escreve” serve para relacionar os livros aos autores que os escreveram (relaciona as entidades autor com a entidade livro). Escolhemos fazer esta decisão em termos de modelo ER, pois o autor não pode ser um atributo, visto que pode existir mais do que um autor e para evitar a redundância de informação repetida nos atributos.

Esta relação é do tipo (N:M), pois um livro pode ter vários autores e um autor pode ter vários livros na biblioteca. É total no lado dos livros, pois tem sempre um autor. É total no lado dos autores, pois todos os livros na biblioteca tem um ou mais autores.



A relação “Trabalha” serve para relacionar os funcionários e o trabalho que eles praticam (relaciona as entidades Funcionário com a entidade Trabalho) .

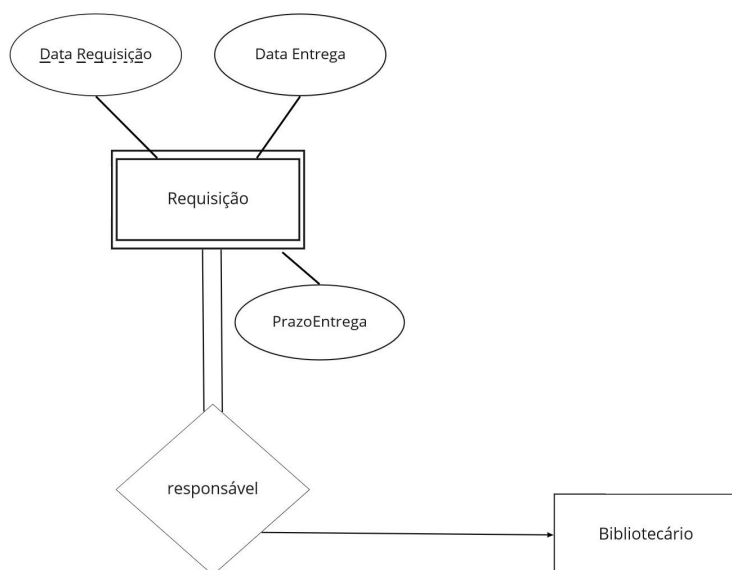
Como trabalhos muito semelhantes podem haver funcionários com a mesma descrição ou ordenado então terão o mesmo trabalho se tiver a mesma descrição e o mesmo ordenado. Esta relação serve para relacionar as entidades “Funcionário” ao seu trabalho e é de (N:1) sendo total no lado da entidade "Funcionário".



A relação "Responsável" serve para ligar a entidade bibliotecário à entidade fraca Requisicao. Esta relação serve para diferenciar os vários levantamentos de livros em que o responsável é o bibliotecário.

A relação é total no lado da entidade Requisicao pois qualquer requisição tem apenas um bibliotecário associado.

Então será uma relação (N:1).



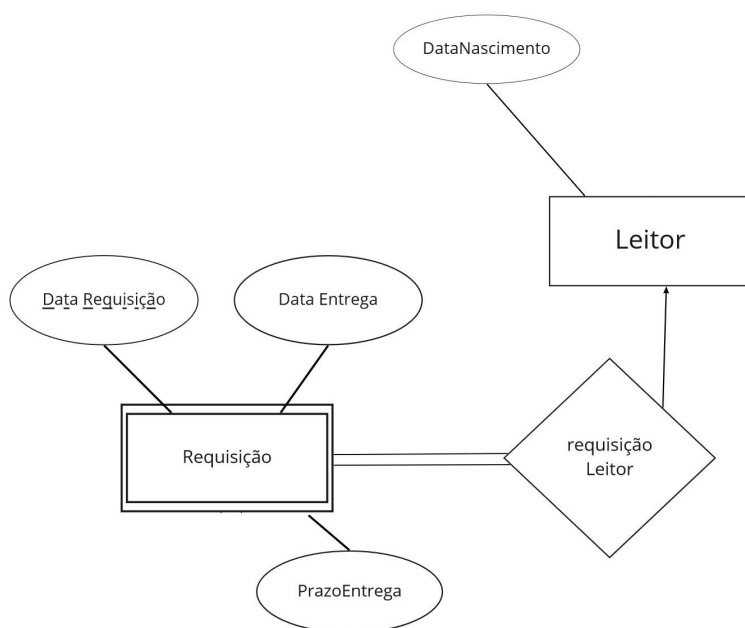
Requisição Leitor:

A relação "requisição Leitor" serve para relacionar a entidade Leitor com a Entidade Fraca requisição.

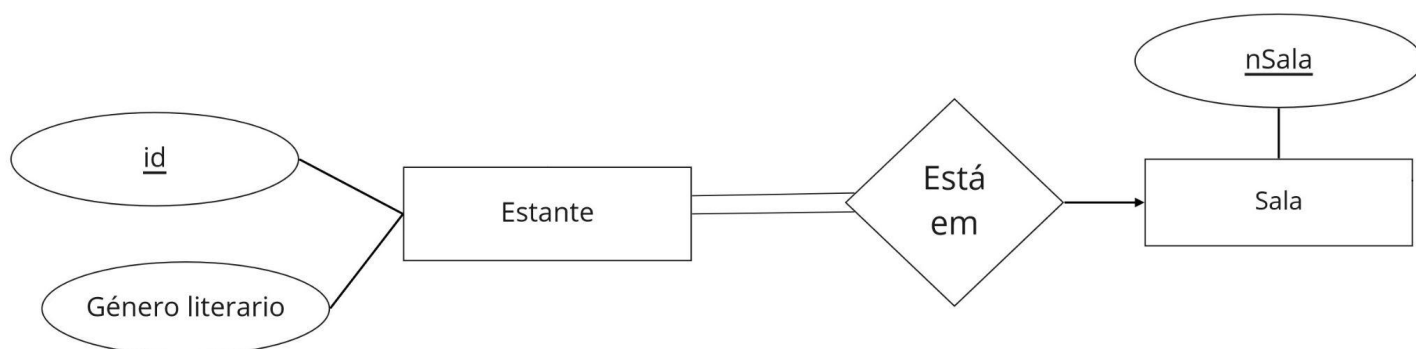
Esta relação serve para identificar quando o Leitor fez levantamentos de livros.

A relação é total no lado da entidade Requisicao pois qualquer requisição tem apenas um Leitor associado.

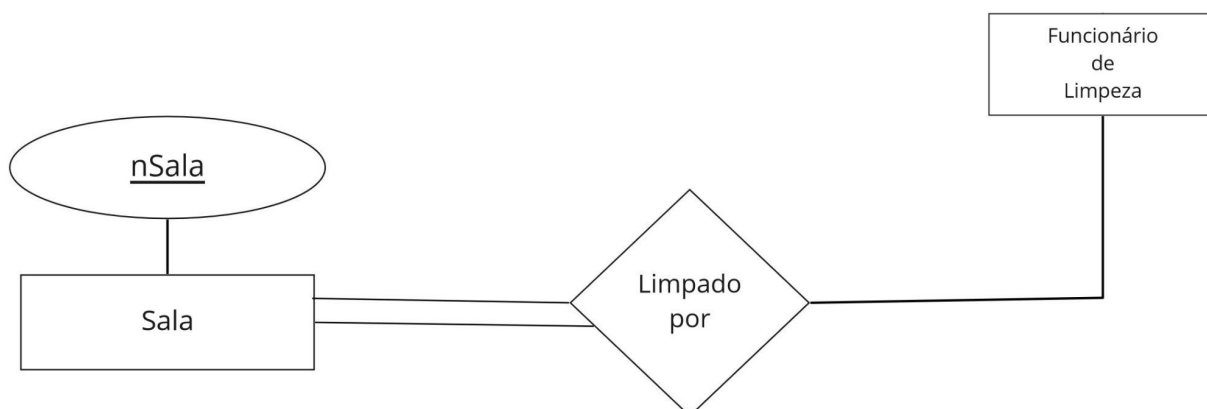
Então será uma relação (N:1).



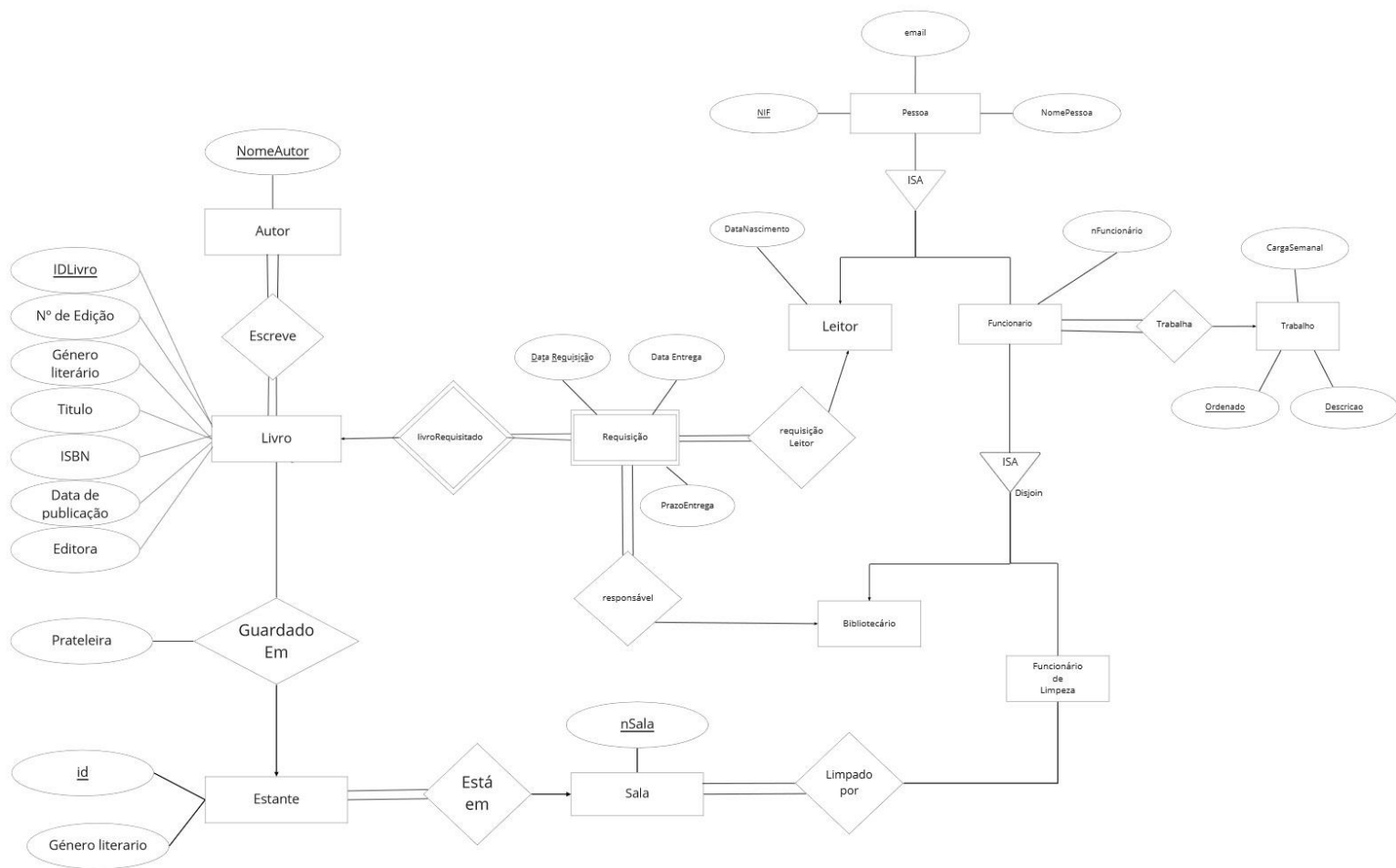
Relação “Está em” relaciona a Entidade Estante e a entidade Sala.
Qualquer estante estará numa sala.
E qualquer sala pode ter um várias estantes.
Então será uma relação (N:1).



Relação “Limpado por” relaciona as entidade Sala e entidade Funcionário de Limpeza.
Cada sala é limpa por um funcionário, mas nem todo o funcionário terá de estar a limpar salas.
Então será uma relação (N:1).



MODELO ER NA TOTALIDADE



MODELO RELACIONAL

TABELAS REDUNDANTES

- Conjuntos de relações muitos-para-um e um-para-muitos, totais no lado *muitos* podem ser representados adicionando atributos extra ao lado muitos contendo a chave primária do outro conjunto participante.
- Se a participação é **parcial** no lado muitos, a substituição da relação (tabela) por atributos extra pode levar à ocorrência de **valores nulos**.
- Para conjuntos de relações um-para-um, qualquer dos lados com participação total pode receber a chave primária do outro lado.
- São redundantes as relações (tabelas) correspondentes aos conjuntos de relações entre o conjunto de entidades fracas e os seus conjuntos de entidades dominantes.

Pessoa (NIF, nomePessoa, email) Chave primária - NIF

Autor (nomeAutor) Chave primária - nomeAutor

Leitor (NIF,DataNascimento) NIF é chave estrangeira referente a pessoa

Funcionario (NIF, nFuncionario,Ordenado,Descricao)

NIF é chave estrangeira referente a pessoa

Ordenado é chave estrangeira referente a Trabalho

Descricao é chave estrangeira referente a Trabalho

Trabalho (Descricao, ordenado, cargaSemanal)

Livro (IDLivro, ISBN , titulo, generoLit , dataPublicacao , editora, Edição)

Chave primária - IDLivro

Estante (id, generoLit, nSala) id é chave primária

nSala é chave estrangeira referente a Sala

Bibliotecario (NIF) NIF é chave estrangeira referente ao funcionário

FuncLimpeza (NIF) NIF é chave estrangeira referente ao funcionário

Escreve(IDLivro,NomeAutor) IDLivro é chave estrangeira que refere Livro

NomeAutor é chave estrangeira que refere Autor

IDLivro e NomeAutor é a super-chave desta relação (chave primária)

Requisicao (DataRequisicao,DataEntrega,PrazoEntrega,Bibliotecario.NIF,Leitor.NIF,IDLivro)

DataRequisicao é chave primária

Bibliotecario.NIF é chave estrangeira referente a Bibliotecario

Leitor.NIF é chave estrangeira referente a Leitor

IDLivro é chave estrangeira referente a Livro

GuardadoEm(IDLivro, id ,prateleira)

IDLivro é chave estrangeira referente a Livro

id é chave estrangeira referente a Estante

LimpadoPor(nSala, NIF)

nSala é chave estrangeira referente a Sala

NIF é chave estrangeira referente a FuncLimpeza

CONSULTAS

Quais são os títulos dos livros da nossa biblioteca?

Quantos autores existem na base de dados da biblioteca?

Qual a estante com mais livros? (id, género literário)

Quantos leitores nascidos antes de 1980 requisitaram livros?

Qual é o autor com o maior número de livros? (nome)

Quantos funcionários existem na nossa biblioteca?

Qual é o bibliotecário que fez maior número de operações (requisições de livros)?
(nome,NIF,nFuncionário)

Quais os livros que nunca foram requisitados? (IDLivro,ISBN, título, editora)

Quais os funcionários que já requisitaram livros da biblioteca?

(nome,NIF, nFuncionario,email)

Quais os funcionários existem com ordenado superior a 1100 euros? Apresente a resposta por ordem decrescente. (nome,NIF, nFuncionario, ordenado, cargaSemanal)

Qual é o funcionário com a maior carga semanal na nossa biblioteca? (nome, nFuncionario, cargaSemanal)

Qual é o livro com maior número de autores? (IDLivro,ISBN, título)

Triggers

O trigger Criar Leitor serve para verificar se existe uma pessoa com o NIF que usamos para criar um Leitor e se não existir , o trigger irá dar uma mensagem de erro:” Nao existe pessoa”

```
CREATE OR REPLACE TRIGGER criarLeitor
BEFORE INSERT OR UPDATE ON Leitor
FOR EACH ROW
DECLARE
    cnt NUMBER;
BEGIN
    SELECT COUNT(NIF) INTO cnt FROM Pessoas WHERE Pessoas.NIF = :NEW.NIFLEITOR;
    IF (cnt = 0)
    THEN Raise_Application_Error (-20101, 'Nao existe pessoa');
    END IF;
END;
/
```

O trigger criarFuncionario serve para verificar se existe uma pessoa com o NIF que usamos para criar um Funcionário e se não existir , o trigger irá dar uma mensagem de erro:” Nao existe pessoa”

```
CREATE OR REPLACE TRIGGER criarFuncionario
BEFORE INSERT OR UPDATE ON Funcionario
FOR EACH ROW
DECLARE
    cnt NUMBER;
BEGIN
    SELECT COUNT(NIF) INTO cnt FROM Pessoas WHERE Pessoas.NIF = :NEW.NIFFUNCIONARIO;
    IF (cnt = 0)
    THEN Raise_Application_Error (-20101, 'Nao existe pessoa');
    END IF;
END;
/
```

O trigger criarFuncLimpeza serve para verificar se já existe uma pessoa com o NIF na tabela dos bibliotecarios. Se existir, irá aparecer uma mensagem de erro, pois as tabelas Bibliotecário e FuncLimpeza são tabelas disjuntas.

Logo não pode haver o mesmo NIF em tabelas diferentes.

```
CREATE OR REPLACE TRIGGER criarFuncLimpeza
BEFORE INSERT OR UPDATE ON FuncLimpeza
FOR EACH ROW
DECLARE
    cnt NUMBER;
BEGIN
    SELECT COUNT(NIFBIBLIOTECARIO) INTO cnt FROM Bibliotecario WHERE Bibliotecario.NIFBIBLIOTECARIO = :NEW.NIFFUNCLIMPEZA;
    IF (cnt > 0)
    THEN Raise_Application_Error (-20101, 'Pessoa nao pode ser funcionario de limpeza, pois ja e bibliotecario');
    END IF;
END;
/
```

O trigger CriarBibliotecario serve para verificar se já existe uma pessoa com o NIF na tabela dos funcionários. Se existir, irá aparecer uma mensagem de erro, pois as tabelas Bibliotecário e FuncLimpeza são tabelas disjuntas. Logo não pode haver o mesmo NIF em tabelas diferentes.

```
CREATE OR REPLACE TRIGGER criarBibliotecario
BEFORE INSERT OR UPDATE ON Bibliotecario
FOR EACH ROW
DECLARE
    cnt NUMBER;
BEGIN
    SELECT COUNT(NIFFUNCLIMPEZA) INTO cnt FROM FuncLimpeza WHERE FuncLimpeza.NIFFUNCLIMPEZA = :NEW.NIFBIBLIOTECARIO;
    IF (cnt > 0)
    THEN Raise_Application_Error (-20101, 'Pessoa nao pode ser bibliotecario, pois ja e funcionario de limpeza');
    END IF;
END;
/
```

O Trigger requisitarLivro, que é ativado antes de qualquer requisição ser registada, foi criado com o intuito de verificar se o livro a ser requisitado está, de facto, disponível na biblioteca para tal ação, consultando a tabela GuardadoEm. Além disto, um leitor que ainda não tenha entregado um livro após o seu prazo de entrega, (data estabelecida no momento da requisição) está proibido de realizar qualquer outra requisição, aparecendo a seguinte mensagem de erro: “O leitor está castigado, porque tem livros por entregar”. Se a requisição for bem sucedida, (sem exceções levantadas e antes ainda da criação da requisição) o livro é apagado da tabela GuardadoEm, pois deixará de existir na estante da biblioteca.

```
CREATE OR REPLACE TRIGGER requisitarLivro
BEFORE INSERT ON Requisicao
FOR EACH ROW
DECLARE
    cnt NUMBER;
    castigo NUMBER;
BEGIN
    SELECT COUNT(*) INTO cnt FROM GuardadoEm WHERE :NEW.IDLivro = GuardadoEm.IDLivro;
    IF (cnt = 0)
    THEN Raise_Application_Error (-20101, 'Livro nao disponivel para requisicao');
    END IF;
    SELECT COUNT(*) INTO castigo FROM Requisicao WHERE (:NEW.DataRequisicao > Requisicao.PrazoEntrega
    and Requisicao.NIFLeitor = :NEW.NIFLeitor and Requisicao.DataEntrega IS NULL);
    IF (castigo > 0)
    THEN Raise_Application_Error(-20101, 'O leitor esta castigado, porque tem livros por entregar');
    END IF;
    DELETE FROM GuardadoEm WHERE :NEW.IDLivro = GuardadoEm.IDLivro;
END;
/
```

O trigger devolverOLivro irá verificar se o atributo data de entrega é Null ou não o é.

Se o atributo dado for Null , irá aparecer mensagem de erro, pois não faz sentido um livro ter sido devolvido ou atualizado sem data de entrega .

```
CREATE OR REPLACE TRIGGER devolverOLivro
BEFORE Update ON Requisicao
FOR EACH ROW
BEGIN
IF (:NEW.DataEntrega IS NULL)
    THEN Raise_Application_Error(-20101, 'Atribuicao invalida');
END IF;
INSERT INTO GuardadoEm (IDESTANTE, IDLIVRO, PRATELEIRA)
VALUES(0, :NEW.IDLivro, 0);
END;
/
```

O trigger arrumarNaEstante irá verificar se o Livro que estamos a colocar na estante é do mesmo género literário da estante, para facilitar a organização da biblioteca, ou então se a Estante é a Estante de arrumar os Livros (com id=0).

Se não acontecer qualquer uma das condições vai aparecer mensagem de erro para não arrumar livros: “O género literário não é o mesmo”.

```
CREATE OR REPLACE TRIGGER arrumarNaEstante
BEFORE INSERT ON GuardadoEm
FOR EACH ROW
DECLARE
generoLivro VARCHAR2(30);
generoEstante VARCHAR2(30);
BEGIN
SELECT GENEROLITERARIO INTO generoLivro FROM LIVRO WHERE Livro.IDLivro = :NEW.IDLivro;
SELECT GENEROLITERARIO INTO generoEstante FROM ESTANTE WHERE Estante.IDEstante = :NEW.IDEstante;

IF(generoLivro <> generoEstante and :NEW.IDEstante <> 0)
    THEN Raise_Application_Error(-20101, 'O Genero Literario nao e o mesmo');
END IF;
END;
/
```

O trigger rearrumarNaEstante serve para colocar os livros de uma estante noutra.

Apenas é permitido colocar livros em estantes quando ambos são do mesmo género literário.

E na recolocação de livros, estes não podem ir para a estante 0, porque não é permitido desarrumar livros.

```
CREATE OR REPLACE TRIGGER rearrumarNaEstante
BEFORE UPDATE ON GuardadoEm
FOR EACH ROW
DECLARE
generoLivro VARCHAR2(30);
generoEstante VARCHAR2(30);
BEGIN
SELECT GENEROLITERARIO INTO generoLivro FROM LIVRO WHERE Livro.IDLivro = :NEW.IDLivro;
SELECT GENEROLITERARIO INTO generoEstante FROM ESTANTE WHERE Estante.IDEstante = :NEW.IDEstante;

IF (:NEW.IDEstante = 0)
THEN Raise_Application_Error(-20101, 'Atualizacao invalida: Espaco reservado a livros por arrumar');
END IF;

IF(generoLivro <> generoEstante)
THEN Raise_Application_Error(-20101, 'O Genero Literario nao e o mesmo');
END IF;
END;
/
```


Discussão de limitações

Analisando criticamente a funcionalidade prática da base de dados, chegámos à conclusão que uma das limitações presentes no nosso trabalho prende-se com facto de que uma devolução não tem um funcionário associado, levando a que não exista informação na nossa base de dados sobre quem é que foi o responsável pela transação da devolução.

Esta foi uma decisão nossa, pois pensamos e debatemos sobre esta situação.. A razão fundamental pela qual temos o envolvimento do funcionário da biblioteca numa requisição. é pelo facto de alguém ser responsável pelo livro, (pois este estaria em boas condições antes de ser emprestado) ou seja assumimos, em termos práticos, que o funcionário da biblioteca é um funcionário “exemplar”, logo quando faz o empréstimo do livro, este não se encontra danificado, visto que este tem o dever de o verificar antes. No caso de uma devolução, não teríamos de guardar o funcionário que o iria receber e, caso o livro estivesse danificado, quando estivesse na estante por arrumar quem seria responsável pelos danos causados seria o leitor da última requisição (assumindo que o bibliotecário entregou o livro em condições).

Consultas interessantes em SQL

```
-- CONSULTING EXAMPLES

--Quais sao os titulos dos livros da nossa biblioteca?
SELECT TITULO FROM LIVRO;

--Quantos autores existem na base de dados da biblioteca?
SELECT COUNT(*) AS NUMERO_DE_AUTORES FROM AUTOR;

--Qual a estante com mais livros? (id, genero literario)
SELECT IDESTANTE, GENEROLITERARIO, COUNT(*) AS NUMERO_DE_LIVROS
FROM GUARDADOEM INNER JOIN LIVRO ON GUARDADOEM.IDLIVRO = LIVRO.IDLIVRO
GROUP BY IDESTANTE, GENEROLITERARIO
ORDER BY NUMERO_DE_LIVROS DESC;

--Quantos leitores nascidos antes de 1980 requisitaram livros?
SELECT DISTINCT COUNT(*) AS LEITORES_NASCIDOS_ANTES_DE_1980
FROM LEITOR INNER JOIN REQUISICAO ON REQUISICAO.nifleitor = leitor.nifleitor
WHERE to_date('01/01/1980', 'MM/DD/RRRR') > DATANASCIMENTO;

--Qual e o autor com o maior numero de livros? (nome)
SELECT NOMEAUTOR, COUNT (*) AS NUMERO_DE_LIVROS_ESCRITOS
FROM ESCREVE
GROUP BY NOMEAUTOR
ORDER BY NUMERO_DE_LIVROS_ESCRITOS DESC;

--Quantos funcionarios existem na nossa biblioteca?
SELECT COUNT(*) AS NUMERO_DE_FUNCIONARIOS
FROM FUNCIONARIO;

--Qual e o bibliotecario que fez maior numero de operacoes (requisicoes de livros)? (nome,NIF,nFuncionario)
SELECT NOMEPESSOA, NIFFUNCIONARIO, NFUNCIONARIO, COUNT(*) AS NUMERO_DE_REQUISICOES
FROM PESSOAS INNER JOIN (SELECT NFUNCIONARIO, NIFFUNCIONARIO
FROM FUNCIONARIO INNER JOIN REQUISICAO ON REQUISICAO.NIFBIBLIOTECARIO = FUNCIONARIO.NIFFUNCIONARIO)
ON PESSOAS.NIF = NIFFUNCIONARIO
GROUP BY NOMEPESSOA, NIFFUNCIONARIO, NFUNCIONARIO ORDER BY NUMERO_DE_REQUISICOES DESC;

--Quais os livros que nunca foram requisitados? (IDLIVRO, ISBN, titulo, editora)
SELECT IDLIVRO, ISBN, TITULO, EDITORA
FROM ((SELECT IDLIVRO, ISBN, TITULO, EDITORA FROM LIVRO)
MINUS
(SELECT LIVRO.IDLIVRO, ISBN, TITULO, EDITORA
FROM REQUISICAO INNER JOIN LIVRO ON LIVRO.IDLIVRO = REQUISICAO.IDLIVRO));

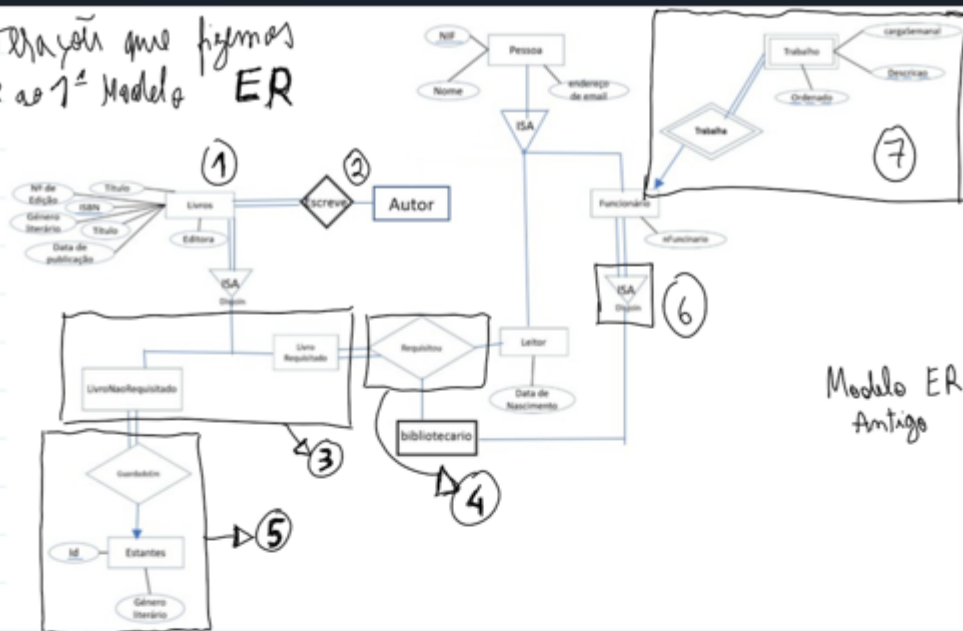
--Quais os funcionarios que ja requisitaram livros da biblioteca? (nome,NIF, nFuncionario,email)
SELECT NOMEPESSOA, NIF, NFUNCIONARIO, MAIL
FROM PESSOAS INNER JOIN (SELECT NFUNCIONARIO, NIFLEITOR
FROM FUNCIONARIO INNER JOIN LEITOR ON NIFLEITOR = NIFFUNCIONARIO) ON NIFLEITOR = PESSOAS.NIF;

--Quais os funcionarios existem com ordenado superior a 1100 euros?
--Apresente a resposta por ordem decrescente. (nome,NIF, nFuncionario, ordenado, cargaSemanal)
SELECT NOMEPESSOA, NIF, NFUNCIONARIO, ORDENADO, CARGASEMANAL
FROM PESSOAS INNER JOIN (SELECT NFUNCIONARIO, NIFFUNCIONARIO, CARGASEMANAL, FUNCIONARIO.ORDENADO
FROM FUNCIONARIO INNER JOIN TRABALHO ON FUNCIONARIO.ORDENADO = TRABALHO.ORDENADO
AND FUNCIONARIO.DESCRICAO = TRABALHO.DESCRICAO) ON NIFFUNCIONARIO = PESSOAS.NIF
WHERE ORDENADO > 1100
ORDER BY ORDENADO DESC;

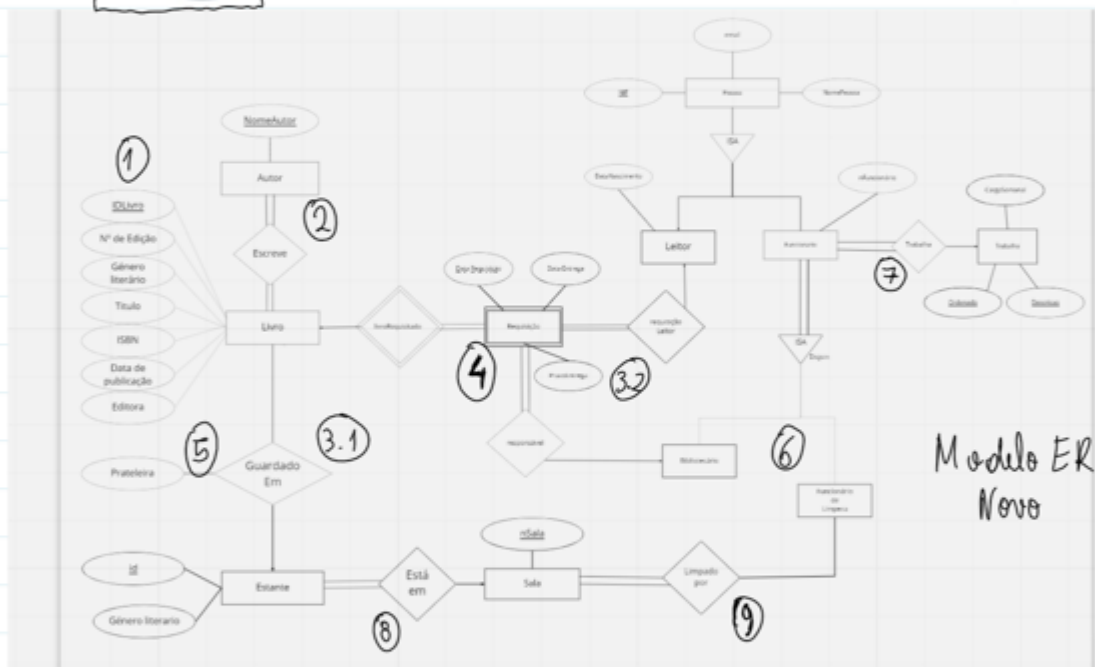
--Qual e o funcionario com a maior carga semanal na nossa biblioteca? (nome, nFuncionario, cargaSemanal)
SELECT NOMEPESSOA, NIF, NFUNCIONARIO, ORDENADO, CARGASEMANAL
FROM PESSOAS INNER JOIN (SELECT NFUNCIONARIO, NIFFUNCIONARIO, CARGASEMANAL, FUNCIONARIO.ORDENADO
FROM FUNCIONARIO INNER JOIN TRABALHO ON FUNCIONARIO.ORDENADO = TRABALHO.ORDENADO
AND FUNCIONARIO.DESCRICAO = TRABALHO.DESCRICAO) ON NIFFUNCIONARIO = PESSOAS.NIF
GROUP BY NOMEPESSOA, NIF, NFUNCIONARIO, ORDENADO, CARGASEMANAL
ORDER BY CARGASEMANAL DESC;

--Qual e o livro com maior numero de autores? (IDLIVRO, ISBN, TITULO)
SELECT LIVRO.IDLIVRO, LIVRO.ISBN, LIVRO.TITULO, COUNT (*) AS NUMERO_DE_AUTORES_DO_LIVRO
FROM LIVRO INNER JOIN ESCREVE ON LIVRO.IDLIVRO = ESCREVE.IDLIVRO
GROUP BY LIVRO.IDLIVRO, LIVRO.ISBN, LIVRO.TITULO
ORDER BY NUMERO_DE_AUTORES_DO_LIVRO DESC;
```

Alterações que fizemos
face ao 1º Modelo ER



Modelo ER Antigo



Modelo ER Novo

- ① Antes o ISBN não identificava um livro específico portanto adicionamos um atributo IDLivro que é a chave primária de Livro
- ② A relação autor passou a ser total no lado do autor
- ③ Alteramos a divisão de livro não guardado na Biblioteca e o que está guardado lá através da relação GuardadoEm ③.1 conseguimos ver quais são os livros que não estão guardados na biblioteca, se em qualquer Requisito de um livro o atributo DataEntrada=null então está estocado
- ④ A mesma relação requisições estava mal pois não permitia que uma pessoa fizera duas requisições do mesmo livro pelo mesmo bibliotecário então mudamos para uma entidade para cuja chave primária é dataRequisição e IDLivro, a DataEntrada começa a null e é diferente de null depois de ser desenvolvido o livro.
É total para leitor e para bibliotecário também faz requisição de um livro e de um bibliotecário. (O responsável pela entrega é irrelevante para a nossa base de dados).

⑤ Adicionamos o atributo 'prateleira' à relação quando o professor indicava-nos para especificarmos onde o livro estava na Estante

⑥ Adicionamos mais uma entidade funcionario de limpeza após críticas do professor pois como antes o ISA total dispunha que minha de Funcionario ia só para bibliotecario, (fizemos isso de forma a que fosse fácil adicionar mais tipos diferentes de funcionarios no futuro o que o professor disse que não estava bem), então criamos uma nova entidade funcionario de Limpeza devido também ao feedback do professor (que o trabalho estava simplista).

⑦ Resolvemos o mesmo problema com a relação trabalha que não era uma entidade pois da mesma forma conseguimos encontrar os trabalhos que estão disponíveis

⑧ ⑨ Acrescentamos devido às críticas do professor pelo facto do nosso trabalho estar muito simplista.

Descrição da interface e respetiva explicação

(este tópico cobre três pontos do relatório)

- Uma secção contendo a descrição da interface;
- Uma secção explicando onde/como está implementada cada uma das funcionalidades da interface indicadas na secção Enunciado, bem como outras funcionalidades que achem relevantes para a avaliação.
- Um pequeno manual do utilizador, que explique como usar a base de dados. Esse manual deverá incluir a indicação precisa de uma sequência de passos/operações que demonstre as funcionalidades da interface indicadas na secção Enunciado, bem como outras funcionalidades que achem relevantes para a avaliação.

Os triggers estão todos implementados em SQL Workshop

Recently Created Tables		Recent SQL Commands	
AUTOR	63 minutes ago	create or replace trigger "FUNCIONA...	5 hours ago
TRABALHO	63 minutes ago	CREATE OR REPLACE TRIGGER rearrumar...	8 hours ago
ESCREVE	63 minutes ago	CREATE OR REPLACE TRIGGER arrumarNa...	8 hours ago
ESTANTE	63 minutes ago	CREATE OR REPLACE TRIGGER devolverO...	8 hours ago
FUNCIONARIO	63 minutes ago	CREATE OR REPLACE TRIGGER requisita...	8 hours ago
FUNCLIMPEZA	63 minutes ago	CREATE OR REPLACE TRIGGER criarFunc...	8 hours ago
GUARDADOEM	63 minutes ago	CREATE OR REPLACE TRIGGER criarBibl...	8 hours ago

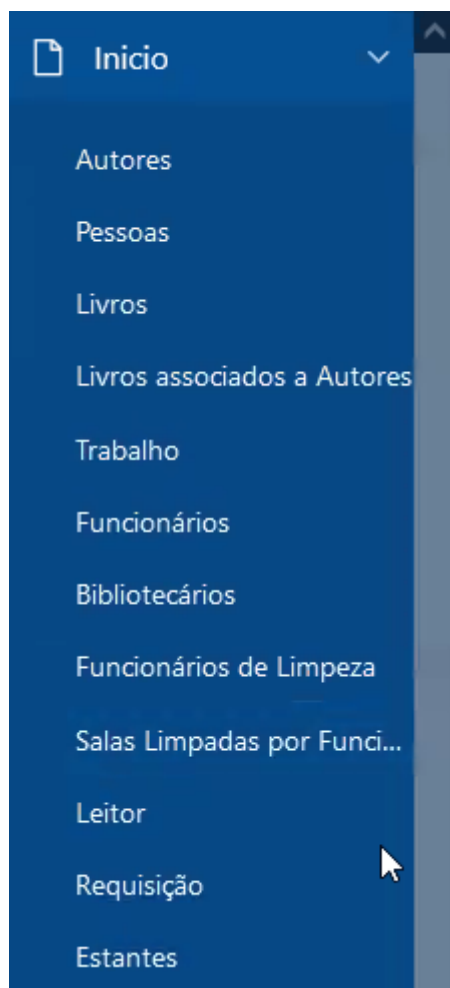
O script que cria as tabelas e insere os valores está em SQL Workshop ->SQL Scripts

<input type="checkbox"/>	Edit	Owner	Name	Created	Updated By	Updated	Bytes	Results	Run
<input type="checkbox"/>		UBD33	Criação de tabelas e Inserção de valores	8 hours ago	UBD33	63 minutes ago	21,229	1	

As relações e as suas operações (criar, remover, alterar) estão implementadas com um “Report with form on Table” - constituído por um “form” e um “interactive report”.

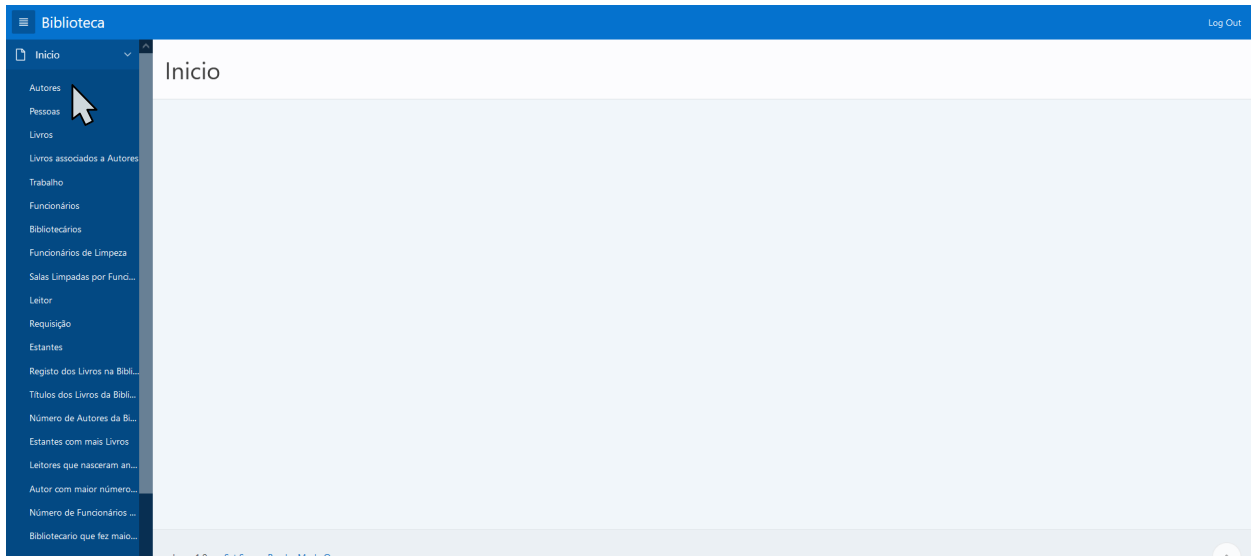
1 - Início	2 - Autores	3 - Criar ou Editar Autores	4 - Pessoas	5 - Criar ou Editar Pessoas	6 - Livros	7 - Criar ou Editar Livros	8 - Livros associados a Autor...
9 - Criar ou Editar associações	10 - Trabalho	11 - Criar ou Editar Trabalho	12 - Funcionários	13 - Criar ou Editar Funcioná...	14 - Bibliotecários	15 - Criar ou Editar Bibliotec...	16 - Funcionários de Limpeza
17 - Criar ou Editar Funcioná...	18 - Salas Limpadas por Fun...	19 - Criar ou Editar Salas Lim...	20 - Leitor	21 - Criar ou Editar Leitor	22 - Requisição	23 - Criar ou Editar Requisição	24 - Estantes
25 - Criar ou Editar Estantes							

Interface das Relações e Entidades, criadas com um “Report with form on Table”

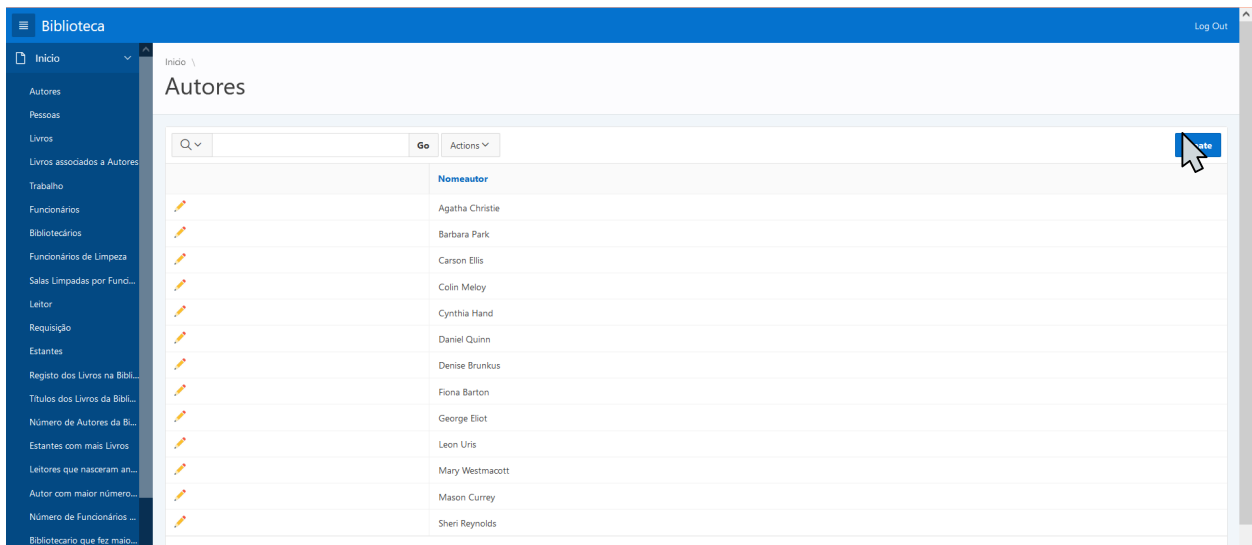


Manual de Utilizador com imagens explicativas

Abrindo a aplicação, é apresentado um menu da base de dados da biblioteca.



Neste menu é possível seleccionar a entidade, relação ou consulta pretendida. No fundo, o registo de todas as informações guardadas na biblioteca. A título de demonstração das funcionalidades da base de dados, seleccionamos a entidade do Autor, presente no menu.



Quando a tabela é apresentada, é possível visualizar os dados já existentes da base de dados, neste caso, a tabela dos Autores, apresentando os seus atributos. É também possível adicionar um elemento à tabela, seleccionando o botão “Create” como representado na imagem.

Biblioteca

Log Out

Início \ Autores \

Criar ou Editar Autores

Form on AUTOR

* Nome do Autor: Sophia de Mello Breyner Andres

Cancel Create

Após o preenchimento dos dados sobre o novo elemento da tabela (tanto opcionais, como obrigatórios), o utilizador pode cancelar a operação pelo botão “Cancel” ou confirmar a inserção do elemento, clicando no botão “Create”.

Biblioteca

✓ Action Processed.

Início \

Autores

Q Go Actions Create

	Nomeautor
	Agatha Christie
	Barbara Park
	Carson Ellis
	Colin Meloy
	Cynthia Hand
	Daniel Quinn
	Denise Brunkus
	Fiona Barton
	George Eliot
	Leon Uris
	Mary Westmacott
	Mason Currey
	Sheri Reynolds
	Sophia de Mello Breyner

Após o último passo, é possível verificar o novo elemento apresentado na tabela e na base de dados, sendo também possível editá-lo ou eliminá-lo, clicando no lápis apresentado no lado esquerdo da tabela.

The screenshot shows the 'Biblioteca' application interface. On the left is a sidebar menu with options: Início, Autores, Pessoas, Livros, Livros associados a Autores, Trabalho, Funcionários, Bibliotecários, Funcionários de Limpeza, Salas Limpadas por Funci..., Leitor, Requisição, and Estantes. The main header is 'Biblioteca' with a 'Log Out' link. The breadcrumb trail is 'Início \ Autores \'. The page title is 'Criar ou Editar Autores'. Below this is a form titled 'Form on AUTOR'. It contains a red asterisk followed by 'Nome do Autor:' and a text input field containing 'Sophia de Mello Breyner'. At the bottom of the form are two buttons: 'Cancel' and 'Delete' (partially visible). To the right of the 'Delete' button is a blue button labeled 'Apply Changes'. A mouse cursor is pointing at the 'Delete' button.

Neste menu é apresentada a opção de remover ou aplicar as mudanças, confirmando-as com o botão de “Delete” ou “Apply Changes”, consoante o utilizador pretenda remover ou editar o elemento, respetivamente.

The screenshot shows the 'Biblioteca' application interface. On the left is a sidebar menu with options: Início, Autores, Pessoas, Livros, Livros associados a Autores, Trabalho, Funcionários, Bibliotecários, Funcionários de Limpeza, Salas Limpadas por Funci..., Leitor, Requisição, and Estantes. The main header is 'Biblioteca' with a 'Log Out' link. The breadcrumb trail is 'Início \ Bibliotecários \'. The page title is 'Criar ou Editar Bibliotecários'. Below this is a form titled 'Form on BIBLIOTECARIO'. It contains a red asterisk followed by 'NIF:' and a text input field containing '225651827'. At the bottom of the form are two buttons: 'Cancel' and 'Create'. A yellow error message box is displayed at the top right of the form area. It contains a warning icon, the text '1 error has occurred', and a list item: 'ORA-20101: Pessoa nao pode ser bibliotecario, pois ja e funcionario de limpeza'. Below the list item is an information icon (i).

No entanto, nem todas as alterações são possíveis, visto que as “regras” do programa têm de ser respeitadas. Neste exemplo, nós tentamos criar um bibliotecário (na tabela dos Bibliotecários), sendo que o identificador deste já é utilizado na tabela dos Funcionários de Limpeza. Uma mensagem de erro (ativada pelo trigger “criarBibliotecario”) é apresentada, explicando que a “Pessoa não pode ser bibliotecário, pois já é funcionário de limpeza, verificando-se assim o “disjoin” das tabelas filhas bibliotecário e funcionário de limpeza, como apresentado no diagrama ER.

Interface das Consultas, criadas com um “interactive report”

Biblioteca

Log Out

Início

Autores

Pessoas

Livros

Livros associados a Autores

Trabalho

Funcionários

Bibliotecários

Funcionários de Limpeza

Salas Limpadas por Funcioná...

Leitor

Requisição

Estantes

Registo dos Livros na Bibliote...

Títulos dos Livros da Biblioteca

Número de Autores da Biblio...

Estantes com mais Livros

Leitores que nasceram antes

Inicio \

Títulos dos Livros da Biblioteca

Q

Go

Actions

Título

Hallowed

Silas Marner: The Weaver of Raveloe

Wildwood

Junie B. Jones and the Stupid Smelly Bus

The Haj

Daily Rituals: How Artists Work

The Rapture of Canaan

Ishmael: An Adventure of the Mind and Spirit

The Widow

Absent in the Spring

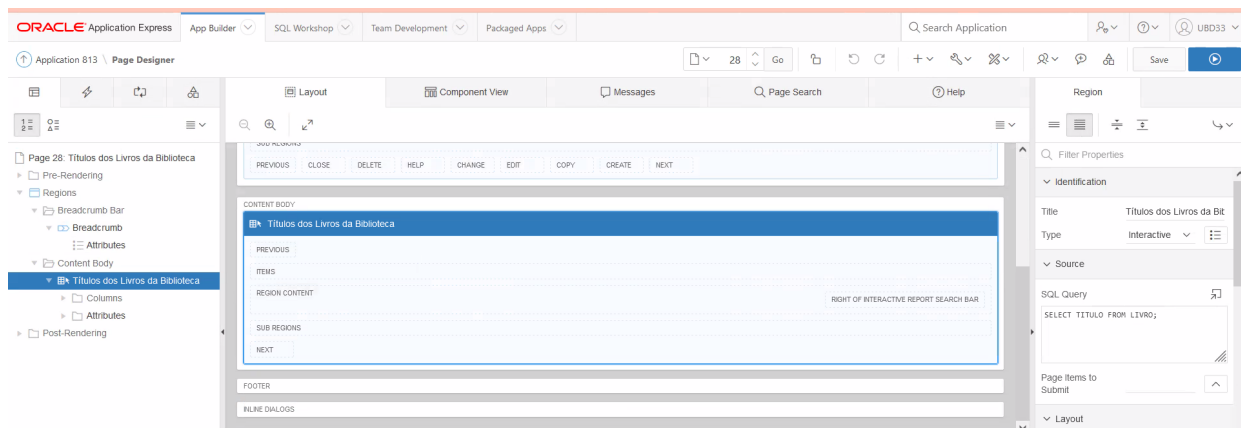
1 - 10

Títulos dos Livros da Bibli...
Número de Autores da Bi...
Estantes com mais Livros
Leitores que nasceram an...
Autor com maior número...
Número de Funcionários ...
Bibliotecario que fez maio...
Livros que nunca foram r...
Funcionários que já requis...
Funcionários com ordena...
Funcionário com a maior ...

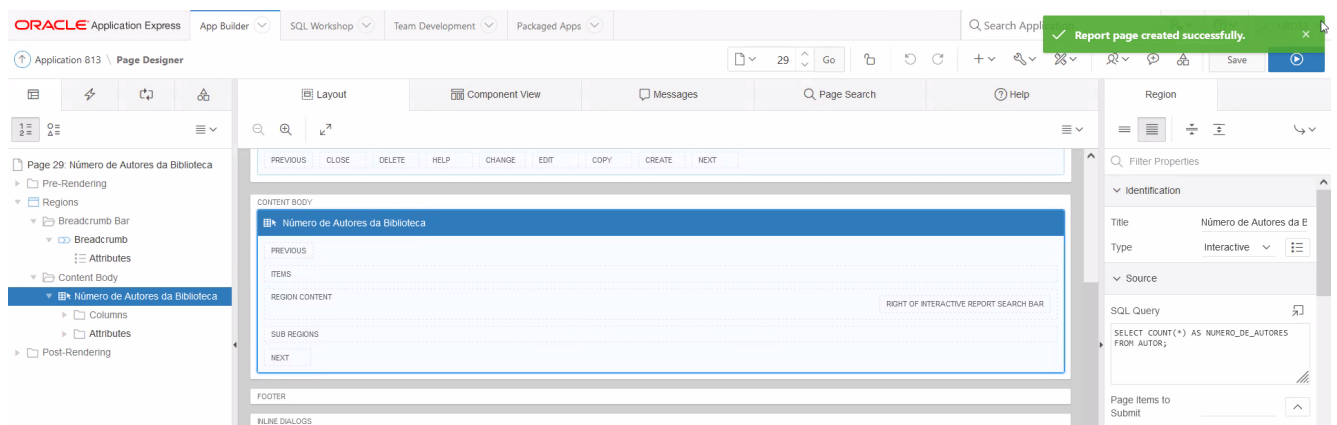
Segue-se o query de cada consulta:

Query em: -> Layout->Region->SQL Query

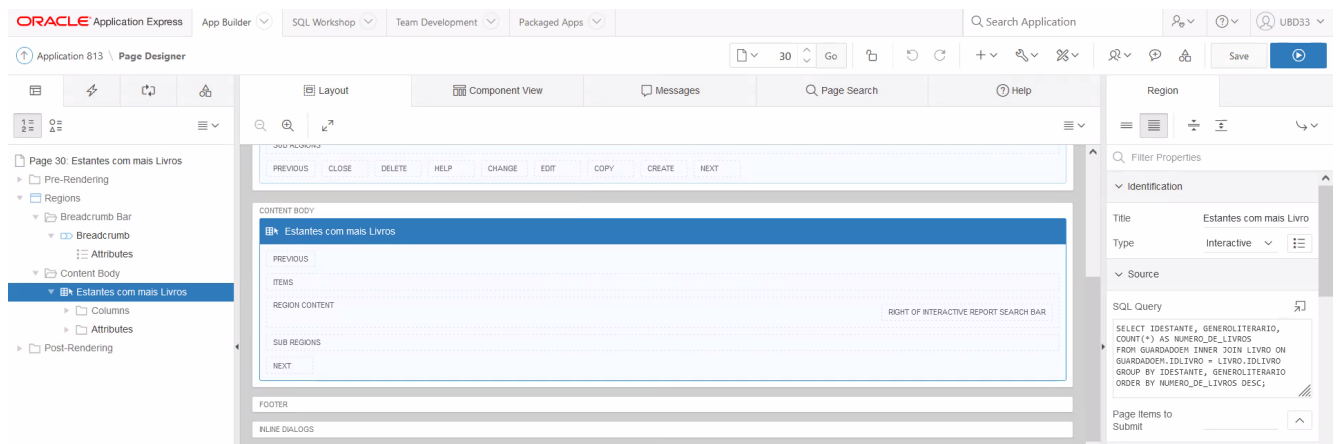
Quais são os títulos dos livros da nossa biblioteca?



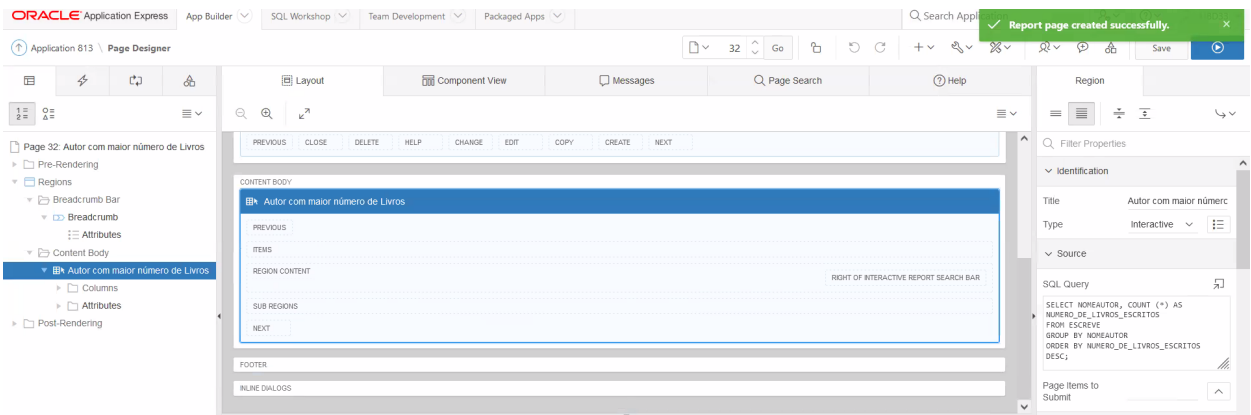
Quantos autores existem na base de dados da biblioteca?



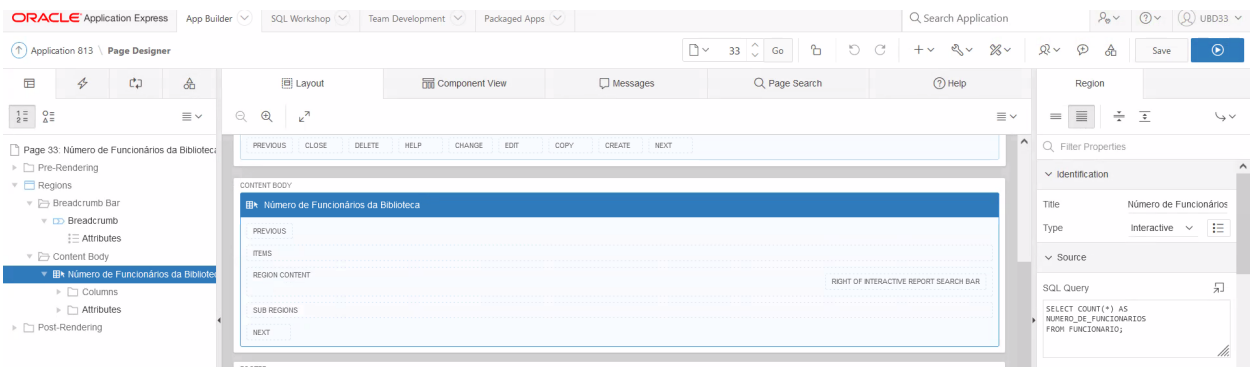
Quais as estantes com mais livros?



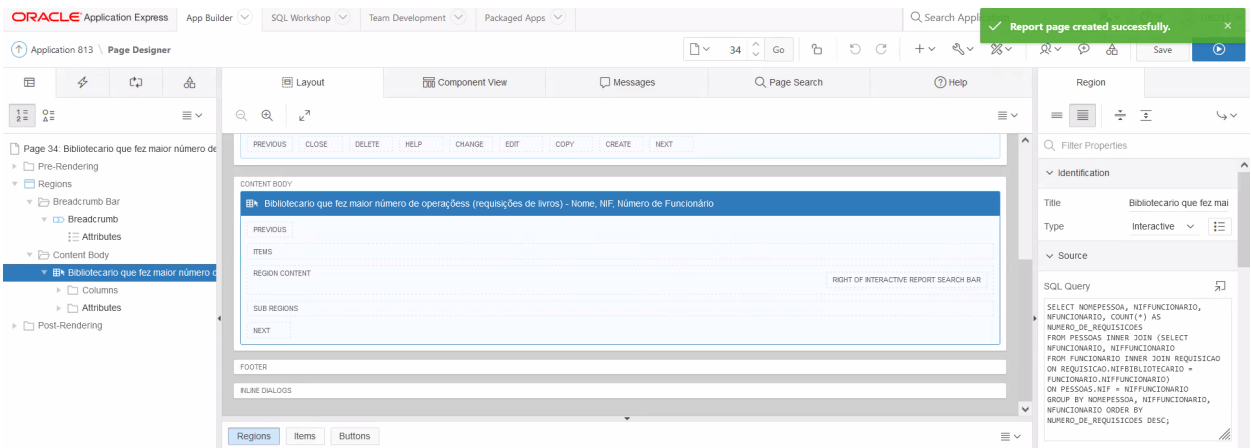
Qual é o autor com o maior número de livros?



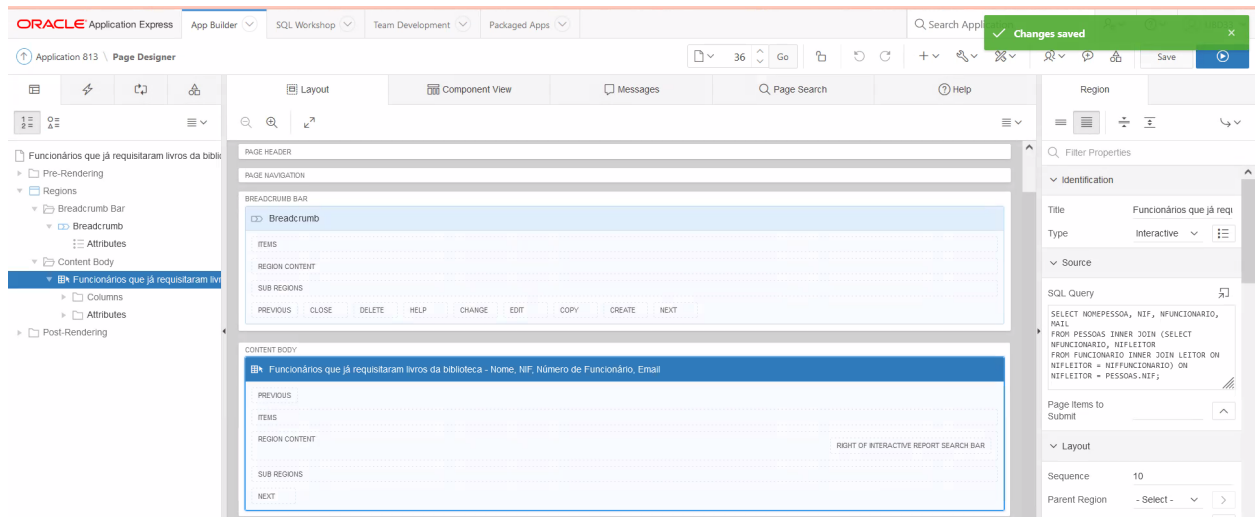
Quantos funcionários existem na nossa biblioteca?



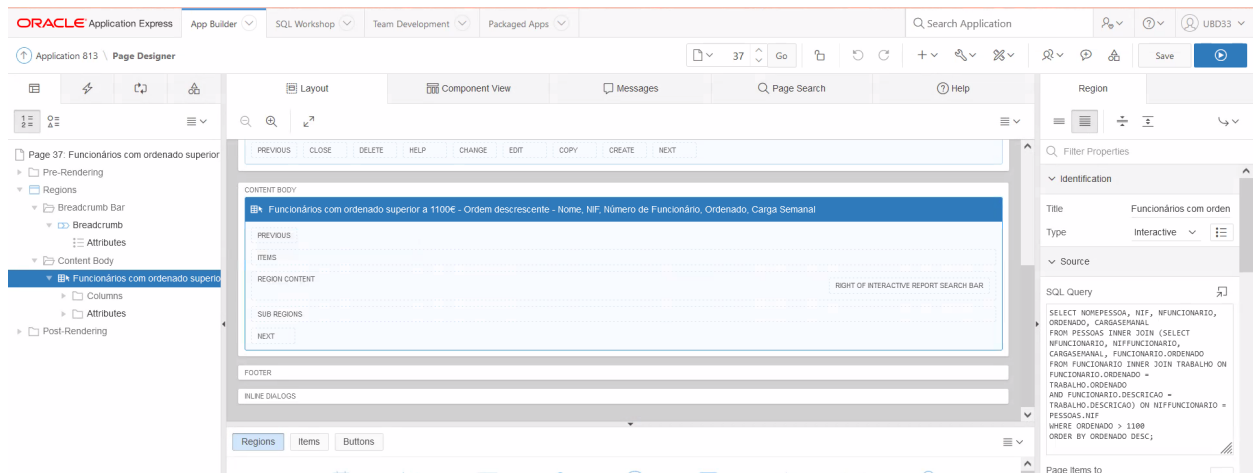
Qual é o bibliotecário que fez maior número de operações (requisições de livros)?



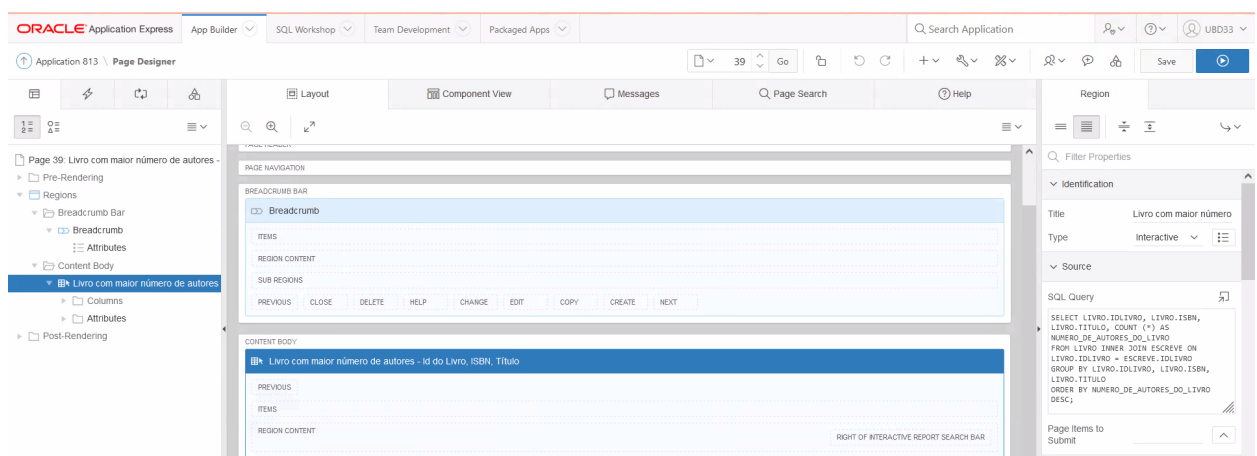
Quais os funcionários que já requisitaram livros da biblioteca?



Quais os funcionários existem com ordenado superior a 1100 euros?



Qual é o livro com maior número de autores?



Dependências funcionais

Através dos identificadores do Livro podemos saber as informações do respetivo livro:

IDLivro → ISbn, titulo, nEdicao, editora, datapublicacao, GeneroLiterario

Com o identificador do livro e o identificador da estante podemos saber em que prateleira o livro está. Pois cada estante tem várias prateleiras, mas só existe uma com aquele livro (se este tiver na biblioteca)

IDLivro, IDEstante → prateleira

Podemos saber qual é o género literário da estante pois cada uma tem o seu género literário.

IDEstante → GeneroLiterario

Através do NIF da pessoa podemos saber o Mail e o nome da pessoa, pois cada pessoa tem um único NIF e cada Nif está associado a um Mail e um Nome.

NIF → Mail, NomePessoa

Através do identificador do livro e da data de requisição do mesmo podemos saber o Nif do bibliotecário, do Nif do leitor e data de entrega do livro. Pois um livro só pode ser requisitado uma vez num certo dia.

IDLivro, DataRequisicao → NIFFuncionario, NIFLeitor, PrazoEntrega, DataEntrega

Através do NIF do funcionário é possível saber a Descrição do trabalho e Ordenado

NIFFuncionario → Ordenado, Descricao

Através do Ordenado do trabalho e a sua descrição é possível saber a carga semanal dos funcionários que desempenham esse trabalho

Ordenado,Descricao→CargaSemanal

Através do Nif do Leitor é possível saber qual é a sua data de nascimento.

NIFLeitor→DataNascimento