# Justificação do Processo de Normalização da Base de Dados

## Introdução

O processo de normalização foi aplicado a um conjunto de dados referente a colaboradores de uma empresa, importado inicialmente como uma única tabela no SQL Server Management Studio (SSMS). Esta tabela continha 35 colunas, incluindo dados como idade, cargo, nível de satisfação, salário, número de anos na empresa, entre outros. Com o objetivo de organizar a informação, reduzir redundâncias e garantir a integridade dos dados, procedeu-se à normalização da estrutura até à Terceira Forma Normal (3FN).

## Primeira Forma Normal (1FN)

Para garantir a 1FN, foi verificado que todas as colunas da tabela original possuíam valores atómicos, ou seja, nenhuma célula continha múltiplos valores. Exemplo: a coluna 'MaritalStatus' contém apenas um estado civil por funcionário ('Single', 'Married' ou 'Divorced'). Não existiam colunas com listas ou campos compostos. Assim, a estrutura original já cumpria este critério.

## Segunda Forma Normal (2FN)

Na segunda etapa, eliminaram-se dependências parciais em relação à chave primária 'EmployeeNumber'. Por exemplo, os campos 'JobRole', 'Department' e 'JobLevel' foram transferidos para a nova tabela 'JobDetails', uma vez que representam o cargo e não atributos diretamente relacionados com o identificador do funcionário. De igual forma, os campos 'Education' e 'EducationField' foram agrupados na tabela 'Education'. Com isso, criaram-se tabelas especializadas, com chaves estrangeiras que apontam para 'EmployeeNumber'.

## Terceira Forma Normal (3FN)

Na 3FN, o foco foi eliminar dependências transitivas. Por exemplo, o campo 'Department' é derivado do 'JobRole'. Logo, o 'Department' não deveria depender diretamente do funcionário, mas sim do cargo. Para resolver isso, foi criada uma tabela auxiliar 'JobRoles', associando cada 'JobRole' ao seu respetivo 'Department'. Outros exemplos incluem a separação de 'MonthlyIncome', 'HourlyRate' e 'DailyRate' na tabela 'Income', e os dados de satisfação ('JobSatisfaction', 'EnvironmentSatisfaction', etc.) na tabela 'Satisfaction'. Cada uma dessas tabelas mantém referência ao funcionário através da chave estrangeira 'EmployeeNumber'.

## Estrutura Final da Base de Dados

Após a normalização, a base de dados foi estruturada nas seguintes tabelas relacionais, cada uma com um conjunto de atributos bem definido:

|  |  |
| --- | --- |
| Tabela | Atributos Principais |
| Employees | EmployeeNumber (PK), Age, Gender, Over18, MaritalStatus |
| Education | EmployeeNumber (FK), Education, EducationField |
| JobDetails | EmployeeNumber (FK), JobRole, JobLevel, BusinessTravel |
| JobRoles | JobRole (PK), Department |
| Income | EmployeeNumber (FK), MonthlyIncome, HourlyRate, DailyRate, MonthlyRate, StockOptionLevel |
| Performance | EmployeeNumber (FK), PerformanceRating, PercentSalaryHike, JobInvolvement |
| Satisfaction | EmployeeNumber (FK), JobSatisfaction, EnvironmentSatisfaction, RelationshipSatisfaction, WorkLifeBalance |
| Tenure | EmployeeNumber (FK), TotalWorkingYears, YearsAtCompany, YearsInCurrentRole, YearsSinceLastPromotion, YearsWithCurrManager |
| EmployeeStatus | EmployeeNumber (FK), Attrition, OverTime |

## Conclusão

A reorganização dos dados em tabelas interligadas permitiu criar uma base de dados mais coerente, sem redundâncias, e com maior capacidade de manutenção e crescimento. A utilização de chaves primárias e estrangeiras garantiu a integridade referencial, enquanto a separação por entidades lógicas facilitou análises futuras e a integração com outros sistemas. O resultado foi uma estrutura mais eficiente, clara e alinhada com os princípios de uma base de dados relacional bem projetada.