**Documentação Power BI, vaga de Analista de Dados**

Autor: Igor Damasceno Mota  
  
Data: 11/06/2023  
  
  
***Tabelas utilizadas:***  
**MICRODADOS\_ENEM\_2020**, tabela importada.

* Fonte: Csv.Document(File.Contents("C:\Users\igord\Downloads\microdados\_enem\_2020\DADOS\MICRODADOS\_ENEM\_2020.csv"),[Delimiter=";", Colunas=76, Encoding=1252, QuoteStyle=QuoteStyle.None]),

**DateTabelaTemplate\_9cfd2296-2188-4799-bdd2-8f052df4a8f5**, tabela importada.

* Fonte: Tabela.FromRows(Json.Document(Binary.Decompress(Binary.FromText("i45WMlTSUfI7vDhfwTk/r7gkUSlWJ1rJCCgWcHhXUk5mMkTAGCRQlFmWmALhmwD5rhUlqUWZ+UVKsbEA", BinaryEncoding.Base64), Compression.Deflate)), let \_t = ((type nullable text) meta [Serialized.Text = true]) in type Tabela [PK = \_t, Nome = \_t]),

**Escola**, tabela criada no modelo.

* Fonte: Tabela.FromRows(Json.Document(Binary.Decompress(Binary.FromText("i45WMlTSUfI7vDhfwTk/r7gkUSlWJ1rJCCgWcHhXUk5mMkTAGCRQlFmWmALhmwD5rhUlqUWZ+UVKsbEA", BinaryEncoding.Base64), Compression.Deflate)), let \_t = ((type nullable text) meta [Serialized.Text = true]) in type Tabela [PK = \_t, Nome = \_t]),

**Cor/Raça**, tabela criada no modelo.

* Fonte: Tabela.FromRows(Json.Document(Binary.Decompress(Binary.FromText("i45WMlDSUfI7vDhfwSU1OSexKDElXylWJ1rJECjsVJSYl5wI5hoBuQFFqSUQnjGIl1iUAuGZAHmOuYlFqTkQvimQ75mXcnhtemoeUCQWAA==", BinaryEncoding.Base64), Compression.Deflate)), let \_t = ((type nullable text) meta [Serialized.Text = true]) in type Tabela [PK = \_t, Nome = \_t]),

***Medidas criadas:***

**1 - MediaNota  
  
Expressão:**

AVERAGEX(MICRODADOS\_ENEM\_2020, (MICRODADOS\_ENEM\_2020[NU\_NOTA\_REDACAO] + MICRODADOS\_ENEM\_2020[NU\_NOTA\_CH] + MICRODADOS\_ENEM\_2020[NU\_NOTA\_CN] + MICRODADOS\_ENEM\_2020[NU\_NOTA\_LC] + MICRODADOS\_ENEM\_2020[NU\_NOTA\_MT]) / 5)  
  
**Dependência:**   
Medida: MediaNota  
Tabela: MICRODADOS\_ENEM\_2020   
Coluna: NU\_NOTA\_CN (Tabela: MICRODADOS\_ENEM\_2020)  
Coluna: NU\_NOTA\_CH (Tabela: MICRODADOS\_ENEM\_2020)  
Coluna: NU\_NOTA\_LC (Tabela: MICRODADOS\_ENEM\_2020)  
Coluna: NU\_NOTA\_MT (Tabela: MICRODADOS\_ENEM\_2020)  
Coluna: NU\_NOTA\_REDACAO (Tabela: MICRODADOS\_ENEM\_2020)

**Dependentes:**Medida: MediaNota por TP\_ESCOLA  
Medida: MaiorMediaPessoa  
Medida: RankMediaNota  
Medida: ESCOLA\_MaxMediaNota  
Medida: MediaNota por Cor/Raça  
Medida: MediaNota por Sexo  
Medida: ESCOLA\_MaxMediaNotaNome

**2 - MediaNota por TP\_ESCOLA  
  
Expressão:**

AVERAGEX(

KEEPFILTERS(VALUES('MICRODADOS\_ENEM\_2020'[TP\_ESCOLA])),

CALCULATE([MediaNota]))  
  
**Dependência:**   
Medida: MediaNota por TP\_ESCOLA  
Tabela: MICRODADOS\_ENEM\_2020  
Coluna: TP\_ESCOLA (Tabela: MICRODADOS\_ENEM\_2020)  
Medida: MediaNota (Tabela: MICRODADOS\_ENEM\_2020)

**Dependentes:**

Medida: MediaNota por TP\_ESCOLA

**3 – MaiorMediaPessoa  
  
Expressão:**

MAXX(

KEEPFILTERS(VALUES('MICRODADOS\_ENEM\_2020'[NU\_INSCRICAO])),CALCULATE([MediaNota]))

**Dependência:**Medida: MaiorMediaPessoa  
Tabela: MICRODADOS\_ENEM\_2020  
Coluna: NU\_INSCRICAO (Tabela: MICRODADOS\_ENEM\_2020)  
Medida: MediaNota (Tabela: MICRODADOS\_ENEM\_2020)

**Dependentes:**

Medida: MaiorMediaPessoa

**4 – RankMediaNota  
  
Expressão:**

RANKX(ALL('MICRODADOS\_ENEM\_2020'[TP\_ESCOLA]), CALCULATE([MediaNota]), , DESC, Dense)

**Dependência:**   
Medida: RankMediaNota  
Tabela: MICRODADOS\_ENEM\_2020  
Coluna: TP\_ESCOLA (Tabela: MICRODADOS\_ENEM\_2020)  
Medida: MediaNota (Tabela: MICRODADOS\_ENEM\_2020)

**Dependentes:**

Medida: RankMediaNota

**5 - ESCOLA\_MaxMediaNota  
  
Expressão:**

CALCULATE(

VALUES(MICRODADOS\_ENEM\_2020[TP\_ESCOLA]),

TOPN(1, ALL('MICRODADOS\_ENEM\_2020'[TP\_ESCOLA]), [MediaNota], DESC) )

**Dependência:**   
Medida: ESCOLA\_MaxMediaNota  
Tabela: MICRODADOS\_ENEM\_2020  
Coluna: TP\_ESCOLA (Tabela: MICRODADOS\_ENEM\_2020)  
Medida: MediaNota (Tabela: MICRODADOS\_ENEM\_2020)

**Dependentes:**Medida: ESCOLA\_MaxMediaNota

**6 - ESCOLA\_MaxMediaNotaNome  
  
Expressão:**

VAR EscolaCodigo =

CALCULATE (

VALUES ( MICRODADOS\_ENEM\_2020[TP\_ESCOLA] ),

TOPN ( 1, ALL ( 'MICRODADOS\_ENEM\_2020'[TP\_ESCOLA] ), [MediaNota], DESC ) )

RETURN

LOOKUPVALUE ( Escola[Nome], Escola[PK], EscolaCodigo )

**Dependência:**   
Medida: ESCOLA\_MaxMediaNotaNome  
Tabela: MICRODADOS\_ENEM\_2020  
Coluna: TP\_ESCOLA (Tabela: MICRODADOS\_ENEM\_2020)  
Medida: MediaNota (Tabela: MICRODADOS\_ENEM\_2020)  
Tabela: Escola  
Coluna: PK (Tabela: Escola)  
Coluna: Nome (Tabela: Escola)

**Dependentes:**

Medida: ESCOLA\_MaxMediaNotaNome

**7 - MediaNotasGeral  
  
Expressão:**   
AVERAGE(MICRODADOS\_ENEM\_2020[MediaEnem])

**Dependência:**   
Medida: MediaNotasGeral  
Tabela: MICRODADOS\_ENEM\_2020  
Calc\_Coluna: MediaEnem (Tabela: MICRODADOS\_ENEM\_2020)

**Dependentes:**

Medida: MediaNotasGeral

**8 - Total Inscritos**

**Expressão:**   
COUNT(MICRODADOS\_ENEM\_2020[NU\_INSCRICAO])

**Dependência:**   
Medida: Total Inscritos  
Tabela: MICRODADOS\_ENEM\_2020  
Coluna: NU\_INSCRICAO (Tabela: MICRODADOS\_ENEM\_2020)

**Dependentes:**

Medida: Total Inscritos

**9 - MediaNota por Cor/Raça**

**Expressão:**

AVERAGEX(

KEEPFILTERS(VALUES('MICRODADOS\_ENEM\_2020'[TP\_COR\_RACA])),

CALCULATE([MediaNota]))

**Dependência:**   
Medida: MediaNota por Cor/Raça  
Tabela: MICRODADOS\_ENEM\_2020  
Coluna: TP\_COR\_RACA (Tabela: MICRODADOS\_ENEM\_2020)  
Medida: MediaNota (Tabela: MICRODADOS\_ENEM\_2020)

**Dependentes:**

Medida: MediaNota por Cor/Raça

**10 - MediaNota por Sexo  
  
Expressão:**

AVERAGEX(

KEEPFILTERS(VALUES('MICRODADOS\_ENEM\_2020'[TP\_SEXO])),

CALCULATE([MediaNota]))

**Dependência:**   
Medida: MediaNota por Sexo  
Tabela: MICRODADOS\_ENEM\_2020  
Coluna: TP\_SEXO (Tabela: MICRODADOS\_ENEM\_2020)  
Medida: MediaNota (Tabela: MICRODADOS\_ENEM\_2020)  
  
**Dependentes:**

Medida: MediaNota por Sexo

**11 - %Ausentes**

**Expressão:**

DIVIDE(

CALCULATE(

COUNTROWS(MICRODADOS\_ENEM\_2020),

MICRODADOS\_ENEM\_2020[NU\_NOTA\_CH] <> 0,

MICRODADOS\_ENEM\_2020[NU\_NOTA\_CN] <> 0,

MICRODADOS\_ENEM\_2020[NU\_NOTA\_LC] <> 0,

MICRODADOS\_ENEM\_2020[NU\_NOTA\_MT] <> 0

),

COUNTROWS(MICRODADOS\_ENEM\_2020))  
  
**Dependência:**Medida: %Ausentes  
Tabela: MICRODADOS\_ENEM\_2020  
Coluna: NU\_NOTA\_CN (Tabela: MICRODADOS\_ENEM\_2020)  
Coluna: NU\_NOTA\_CH (Tabela: MICRODADOS\_ENEM\_2020)  
Coluna: NU\_NOTA\_LC (Tabela: MICRODADOS\_ENEM\_2020)  
Coluna: NU\_NOTA\_MT (Tabela: MICRODADOS\_ENEM\_2020)

**Dependentes:**

Medida: %Ausentes

**12 - Media Ciencias Humanas**

**Expressão:**

AVERAGE(MICRODADOS\_ENEM\_2020[NU\_NOTA\_CH])  
  
**Dependência:**

Medida: Media Ciencias Humanas  
Tabela: MICRODADOS\_ENEM\_2020  
Coluna: NU\_NOTA\_CH (Tabela: MICRODADOS\_ENEM\_2020)

**Dependentes:**Medida: Media Ciencias Humanas

**13 - Media Ciencias Natureza**

**Expressão:**

AVERAGE(MICRODADOS\_ENEM\_2020[NU\_NOTA\_CN])  
  
**Dependência:**   
Tabela: MICRODADOS\_ENEM\_2020  
Coluna: NU\_NOTA\_CN (Tabela: MICRODADOS\_ENEM\_2020)

**Dependentes:**Medida: Media Ciencias da Natureza

**14 - Media Linguagem e Códigos**

**Expressão:**

AVERAGE(MICRODADOS\_ENEM\_2020[NU\_NOTA\_LC])  
  
**Dependência:**   
Tabela: MICRODADOS\_ENEM\_2020  
Coluna: NU\_NOTA\_LC (Tabela: MICRODADOS\_ENEM\_2020)

**Dependentes:**Medida: Media Linguagem e Códigos  
  
**15 – Media Matematica**

**Expressão:**

AVERAGE(MICRODADOS\_ENEM\_2020[NU\_NOTA\_MT])  
  
**Dependência:**   
Tabela: MICRODADOS\_ENEM\_2020  
Coluna: NU\_NOTA\_MT (Tabela: MICRODADOS\_ENEM\_2020)

**Dependentes:**Medida: Media Matematica

**16 - Media Redação  
  
Expressão:**

AVERAGE(MICRODADOS\_ENEM\_2020[NU\_NOTA\_REDACAO])  
  
**Dependência:**   
Tabela: MICRODADOS\_ENEM\_2020  
Coluna: NU\_NOTA\_REDACAO (Tabela: MICRODADOS\_ENEM\_2020)

**Dependentes:**Medida: Media Redação  
  
  
***M Script***

**“MICRODADOS\_ENEM\_2020-f0dfc978-9c04-4334-a74c-c5bc832e6bde”**  
  
let  
  
Fonte = Csv.Document(File.Contents("C:\Users\igord\Downloads\microdados\_enem\_2020\DADOS\MICRODADOS\_ENEM\_2020.csv"),[Delimiter=";", Columns=76, Encoding=1252, QuoteStyle=QuoteStyle.None]),  
  
#"Cabeçalhos Promovidos" = Table.PromoteHeaders(Fonte, [PromoteAllScalars=true]),  
  
#"Tipo Alterado" = Table.TransformColumnTypes(#"Cabeçalhos Promovidos",{{"NU\_INSCRICAO", Int64.Type}, {"NU\_ANO", Int64.Type}, {"TP\_FAIXA\_ETARIA", Int64.Type}, {"TP\_SEXO", type text}, {"TP\_ESTADO\_CIVIL", Int64.Type}, {"TP\_COR\_RACA", Int64.Type}, {"TP\_NACIONALIDADE", Int64.Type}, {"TP\_ST\_CONCLUSAO", Int64.Type}, {"TP\_ANO\_CONCLUIU", Int64.Type}, {"TP\_ESCOLA", Int64.Type}, {"TP\_ENSINO", Int64.Type}, {"IN\_TREINEIRO", Int64.Type}, {"CO\_MUNICIPIO\_ESC", Int64.Type}, {"NO\_MUNICIPIO\_ESC", type text}, {"CO\_UF\_ESC", Int64.Type}, {"SG\_UF\_ESC", type text}, {"TP\_DEPENDENCIA\_ADM\_ESC", Int64.Type}, {"TP\_LOCALIZACAO\_ESC", Int64.Type}, {"TP\_SIT\_FUNC\_ESC", Int64.Type}, {"CO\_MUNICIPIO\_PROVA", Int64.Type}, {"NO\_MUNICIPIO\_PROVA", type text}, {"CO\_UF\_PROVA", Int64.Type}, {"SG\_UF\_PROVA", type text}, {"TP\_PRESENCA\_CN", Int64.Type}, {"TP\_PRESENCA\_CH", Int64.Type}, {"TP\_PRESENCA\_LC", Int64.Type}, {"TP\_PRESENCA\_MT", Int64.Type}, {"CO\_PROVA\_CN", Int64.Type}, {"CO\_PROVA\_CH", Int64.Type}, {"CO\_PROVA\_LC", Int64.Type}, {"CO\_PROVA\_MT", Int64.Type}, {"NU\_NOTA\_CN", Int64.Type}, {"NU\_NOTA\_CH", Int64.Type}, {"NU\_NOTA\_LC", Int64.Type}, {"NU\_NOTA\_MT", Int64.Type}, {"TX\_RESPOSTAS\_CN", type text}, {"TX\_RESPOSTAS\_CH", type text}, {"TX\_RESPOSTAS\_LC", type text}, {"TX\_RESPOSTAS\_MT", type text}, {"TP\_LINGUA", Int64.Type}, {"TX\_GABARITO\_CN", type text}, {"TX\_GABARITO\_CH", type text}, {"TX\_GABARITO\_LC", type text}, {"TX\_GABARITO\_MT", type text}, {"TP\_STATUS\_REDACAO", Int64.Type}, {"NU\_NOTA\_COMP1", Int64.Type}, {"NU\_NOTA\_COMP2", Int64.Type}, {"NU\_NOTA\_COMP3", Int64.Type}, {"NU\_NOTA\_COMP4", Int64.Type}, {"NU\_NOTA\_COMP5", Int64.Type}, {"NU\_NOTA\_REDACAO", Int64.Type}, {"Q001", type text}, {"Q002", type text}, {"Q003", type text}, {"Q004", type text}, {"Q005", Int64.Type}, {"Q006", type text}, {"Q007", type text}, {"Q008", type text}, {"Q009", type text}, {"Q010", type text}, {"Q011", type text}, {"Q012", type text}, {"Q013", type text}, {"Q014", type text}, {"Q015", type text}, {"Q016", type text}, {"Q017", type text}, {"Q018", type text}, {"Q019", type text}, {"Q020", type text}, {"Q021", type text}, {"Q022", type text}, {"Q023", type text}, {"Q024", type text}, {"Q025", type text}})  
  
in  
  
#"Tipo Alterado"  
  
  
**“DateTableTemplate\_9cfd2296-2188-4799-bdd2-8f052df4a8f5-cec90e8d-1d61-453b-bb92-6c94ea41be30”**  
  
Calendar(Date(2015,1,1), Date(2015,1,1))  
  
  
  
**“Escola-8141cc85-990c-4991-9895-621cd495adae”**  
  
let  
  
Fonte = Table.FromRows(Json.Document(Binary.Decompress(Binary.FromText("i45WMlTSUfI7vDhfwTk/r7gkUSlWJ1rJCCgWcHhXUk5mMkTAGCRQlFmWmALhmwD5rhUlqUWZ+UVKsbEA", BinaryEncoding.Base64), Compression.Deflate)), let \_t = ((type nullable text) meta [Serialized.Text = true]) in type table [PK = \_t, Nome = \_t]),  
  
#"Tipo Alterado" = Table.TransformColumnTypes(Fonte,{{"PK", Int64.Type}, {"Nome", type text}})  
  
in  
  
#"Tipo Alterado"  
  
**“Cor/raça-63faa1be-fcc7-45af-9bd8-7c3b0af59fe0”**  
  
let  
  
Fonte = Table.FromRows(Json.Document(Binary.Decompress(Binary.FromText("i45WMlDSUfI7vDhfwSU1OSexKDElXylWJ1rJECjsVJSYl5wI5hoBuQFFqSUQnjGIl1iUAuGZAHmOuYlFqTkQvimQ75mXcnhtemoeUCQWAA==", BinaryEncoding.Base64), Compression.Deflate)), let \_t = ((type nullable text) meta [Serialized.Text = true]) in type table [PK = \_t, Nome = \_t]),  
  
#"Tipo Alterado" = Table.TransformColumnTypes(Fonte,{{"PK", Int64.Type}, {"Nome", type text}})  
in  
#"Tipo Alterado"