

FUNÇÕES DE DATA

O Firebird tem a função embutida EXTRACT, agora alguns problemas mais velhos podem ser resolvidos mais facilmente.

- ☐ Dia da Semana
- ☐ Primeiro e Último Dia do mês
- ☐ Número de dias no mês
- ☐ Semana do ano
- ☐ É Ano bissexto?

1. Dia da Semana

Execute: EXTRACT(WEEKDAY FROM D)

Retorna: 0=domingo, 1=segunda, ... 6=sábado.

Por outro lado, o padrão internacional ISO 8601 especifica que determinadas semanas iniciem na segunda-feira e que é dia 1. Para obter pelo padrão ISO 8601:

Execute: EXTRACT(WEEKDAY FROM D-1)+1

Retorna: 1=segunda, 2=terça, ... 7=quinta.

Exemplo:

```
SELECT D,  
       EXTRACT( WEEKDAY FROM D)          AS AMERICAN,  
       EXTRACT( WEEKDAY FROM D - 1) + 1 AS ISO8601  
FROM T;
```

	D AMERICAN	ISO8601
29-NOV-2001	4	4
30-NOV-2001	5	5
1-DEC-2001	6	6
2-DEC-2001	0	7 <<<< domingo
3-DEC-2001	1	1
4-DEC-2001	2	2
5-DEC-2001	3	3

2. Primeiro e último dia do mês

Primeiro dia do mês:

D - EXTRACT(DAY FROM D) + 1;

Último Dia do mês:

LDM = D - EXTRACT(DAY FROM D) + 32;
LDM = LDM - EXTRACT(DAY FROM LDM);

ou como única expressão:

D - EXTRACT(DAY FROM D) + 32 -
EXTRACT(DAY FROM D - EXTRACT(DAY FROM D) + 32)

Primeiro dia do próximo mês:

```
FDNM = D - EXTRACT(DAY FROM D) + 32;  
FDNM = FDNM - EXTRACT(DAY FROM FDNM) + 1;
```

ou como única expressão:

```
D - EXTRACT(DAY FROM D) + 33 -  
  EXTRACT(DAY FROM D - EXTRACT(DAY FROM D) + 32)
```

Exemplo: Se o dia é D = ' 2002-3-22 ',

```
primeiro dia do mês será ' 2002-3-1 '  
último dia do mês será ' 2002-3-31 '  
primeiro dia do mês que vem será ' 2002-4-1 '
```

3. Número de dias no mês

Número de dias do mês em procedimento armazenado:

```
CREATE PROCEDURE MonthLength (D DATE) RETURNS (ML INTEGER) AS  
DECLARE VARIABLE TMP DATE;  
BEGIN  
  TMP = D - EXTRACT(DAY FROM D) + 32;  
  ML = EXTRACT(DAY FROM (TMP - EXTRACT(DAY FROM TMP)));  
END
```

ou como única expressão:

```
EXTRACT(DAY FROM (D - EXTRACT(DAY FROM D) + 32 -  
  EXTRACT(DAY FROM D - EXTRACT(DAY FROM D) + 32)))
```

4. Semana do ano

O padrão ISO 8601 (para Data e Anotação de Tempo) especifica que determinada semana inicie na segunda-feira, e que a primeira semana de um ano é o que inclui a primeira quinta-feira. (Em outras palavras, a semana pertence ao ano onde está a sua parte maior.) Esta é a fórmula básica:

$$(\text{EXTRACT}(\text{YEARDAY FROM D}) - \text{EXTRACT}(\text{WEEKDAY FROM D-1}) + 7) / 7$$

Pode devolver um valor entre 0 e 53. Às vezes pode ser suficiente, porém se nós quisermos conforme o padrão ISO, nós precisamos fazer alguns ajustes no caso do resultado ser zero ou 53.

Se o resultado for zero, significa que na realidade a semana pertence ao ano anterior. A última semana do ano anterior pode ser 52 ou 53; para descobrir qual, nós repetimos há pouco a mesma fórmula (como último dia do mês passado do ano anterior como parâmetro).

Se o resultado for 53, nós temos que conferir se realmente é 53ª semana, ou se é a primeira semana do próximo ano. 53ª semana só é válida se contiver quinta-feira (exemplo: 31.12. é quinta-feira ou sexta-feira).

```
CREATE PROCEDURE YearWeek (D DATE)  
  RETURNS (WEEK_NO VARCHAR(8)) AS  
DECLARE VARIABLE W INTEGER; /* número da semana */  
DECLARE VARIABLE Y INTEGER; /* ano que a semana pertence */  
BEGIN  
  W = (EXTRACT(YEARDAY FROM D) - EXTRACT(WEEKDAY FROM D-1) + 7) / 7;
```

```

Y = EXTRACT(YEAR FROM D);

IF (W=0) THEN BEGIN
  Y = Y - 1;
  D = D - EXTRACT(YEARDAY FROM D) - 1; / * último dia do ano anterior; D aqui é usado
como variável temporária */
  W = (EXTRACT(YEARDAY FROM D) - EXTRACT(WEEKDAY FROM D-1) + 7) / 7;
  END
ELSE
  IF (W=53 AND 4>EXTRACT(WEEKDAY FROM (D - EXTRACT(DAY FROM D) + 31))) THEN
  BEGIN
    Y = Y + 1;
    W = 1;
    END

/ * Isto é para formatação; você pode preferir W e Y devolver parâmetros ao invés. */
IF (W<10) THEN
  WEEK_NO = '0';
ELSE
  WEEK_NO = '';
WEEK_NO = Y || '/' || WEEK_NO || W;
SUSPEND;
END

```

5. É ano bissexto?

Para descobrir se determinada data pertence a ano bissexto, nós podemos olhar por exemplo ao 59º dia do ano. É 29 de fevereiro se for ano bissexto, ou é 1 de março se ano normal.

Um exemplo de procedimento armazenado:

```

CREATE PROCEDURE Is_LeapYear (D DATE) RETURNS (LY INTEGER) AS
BEGIN
  IF ( 2 = EXTRACT(MONTH FROM (D - EXTRACT(YEARDAY FROM D) + 59)) ) THEN
    LY = 1; /* ano bissexto */
  ELSE
    LY = 0; /* ano normal */
  END

```

Outra procedure exemplo (com ano como parâmetro de entrada):

```

CREATE PROCEDURE Is_LeapYear (Y INTEGER) RETURNS (LY INTEGER) AS
BEGIN
  IF ( 60 = EXTRACT(YEARDAY FROM CAST(Y || '-3-1' AS TIMESTAMP)) ) THEN
    LY = 1; /* ano bissexto */
  ELSE
    LY = 0; /* ano normal */
  END

```

Porque a versão 5 do Interbase não suportava a função EXTRACT, nós há pouco podemos testar se 29 de fevereiro existe:

```

CREATE PROCEDURE Is_LeapYear (Y INTEGER) RETURNS (LY INTEGER) AS
DECLARE VARIABLE D DATE;
BEGIN
  LY = 1;
  D = CAST('29-FEB-' || Y AS DATE);

```

```
WHEN ANY DO LY = 0;  
END
```


Uma expressão bem conhecida para testar ano bissexto é esta:

```
Is_LeapYear := ( ((Year MOD 4) = 0) AND ((Year MOD 100) <> 0) ) OR  
               ((Year DIV 400) = 0);
```

Porém o Firebird não suporta o operador MOD diretamente; ou pode ser implementado com uma UDF ou próprio implementando da expressão:

$X \text{ MOD } Y = X - (X \text{ DIV } Y) * Y$

que não é tão fácil quanto parece ser no princípio, isto por causa de regras diferentes para dividir dentro do Dialeto-1 e Dialeto-3.

<p>Artigo Original:</p> <p>http://www.volny.cz/iprenosil/interbase/ip_ib_code_timestamp.htm</p> <p>Ivan Prenosil - prenosil@ms.anet.cz</p>	
<p>Tradução e adaptação:</p> <p>Antonio Porfírio - (Tony)</p> <p>antonioporfirio@ipdal.com.br tonydmaz@yahoo.com.br</p>	<p>Comunidade Firebird de Língua Portuguesa</p> <p>Visite a Comunidade em:</p> <p>http://www.comunidade-firebird.org</p>
<p>A Comunidade Firebird de Língua Portuguesa foi autorizada pelo Autor do Original para elaborar esta tradução.</p>	