

8-3 データ型

8-3-4 日付／時刻

Query Editor

Query History

1

```
select current_date - interval '1 day' as interval;
```

Data Output

Explain

Messages

Notifications

interval

timestamp without time zone

1

2020-12-15 00:00:00

8-3-6 連番

Query Editor

Query History

1

create table serial_test (id serial, name text);

2

insert into serial_test(name) values('TestA');

3

select * from serial_test;

Data Output

Explain

Messages

Notifications

id

integer

name

text

1

1

TestA

Query Editor

Query History

1

insert into serial_test values (2, 'TestB');

2

select * from serial_test;

Data Output

Explain

Messages

Notifications

	id integer	name text
1	1	TestA
2	2	TestB

Query Editor

Query History

1

insert into serial_test (name) values ('TestC');

2

select * from serial_test;

Data Output

Explain

Messages

Notifications

	<div><div>id</div><div>integer</div></div>	<div><div>name</div><div>text</div></div>
1	1	TestA
2	2	TestB
3	2	TestC

上記では、まず「TestA」を挿入した際に、自動的に1が追加されている。
最後に挿入したときは、直前にid=2があるため、自動的に連番（2）が割り振られている。
つまり、数字がなければ1を振り、直前に数字があれば連番を振るといった仕組みである。

8-3-7 OID

OIDとはPostgreSQL独自のデータ型で、オブジェクト（テーブルやインデックス など）の識別に用いられるIDである。
selectの列名で指定しないと、見ることはできない。

Query Editor Query History

1 `select oid, relname from pg_class limit 3;`

Data Output Explain Messages Notifications

	oid oid	relname name
1	25058	pg_toast_25053_index
2	25059	prefectures_pkey
3	25045	areas

テーブル名やインデックス名をOIDに変換することも可能である。
下記では、「regclass」というデータ型へのキャストを利用して実行している。

Query Editor Query History

1 `select oid, relname from pg_class where oid = 'pg_type'::regclass;`

Data Output Explain Messages Notifications

	oid oid	relname name
1	1247	pg_type

8-3-8 配列

Query Editor Query History

1 `create table array_tbl(
2
3 c1 text[],
4 c2 int[],
5 c3 timestamp[]
6);
7
8 select * from array_tbl;`

Data Output Explain Messages Notifications

	c1 text[]	c2 integer[]	c3 timestamp without time zone[]
--	--------------	-----------------	-------------------------------------

配列の挿入

Query EditorQuery History

1create table array_table(c1 int[]);
2
3insert into array_table values ('{1, 2, 3}'), ('{{1, 2, 3}, {2, 3, 4}}');
4
5select * from array_table;

Data OutputExplainMessagesNotifications

c1
integer[]

1{1,2,3}
2{{1,2,3},{2,3,4}}

配列の検索

Query EditorQuery History

1drop table array_table;
2
3create table array_table (c1 text[]);
4
5insert into array_table values ('{"ABC", "DEF", "GHI"}'), ('{"ZYX", "VUT", "SRQ"}');
6
7select c1[1:2] from array_table;

Data OutputExplainMessagesNotifications

c1
text[]

1{ABC,DEF}
2{ZYX,VUT}

下と見比べると、配列が1～2までの出力となっている。

Query EditorQuery History

1select * from array_table;

Data OutputExplainMessagesNotifications

c1
text[]

1{ABC,DEF,GHI}
2{ZYX,VUT,SRQ}

Query Editor	Query History
<pre>1 select * from array_table where c1[2] = 'VUT'; </pre>	
Data Output	Explain Messages Notifications
<div> <div>c1</div> <div>text[]</div> <div></div> </div>	
1	{ZYX,VUT,SRQ}

列c1のうち、'ABC'が配列の要素のどれか1つである、という条件をもとに行を検索をする。

Query Editor	Query History
<pre>1 select * from array_table where 'ABC' = any(c1);</pre>	
Data Output	Explain Messages Notifications
<div> <div>c1</div> <div>text[]</div> <div></div> </div>	
1	{ABC,DEF,GHI}

配列の更新

c1に'ABC'を持つ行を指定して、その要素の4番目に'JKL'を追加する。

Query Editor	Query History
<pre>1 update array_table set c1[4] = 'JKL' where 'ABC' = any(c1); 2 3 select * from array_table where 'ABC' = any(c1);</pre>	
Data Output	Explain Messages Notifications
<div> <div>c1</div> <div>text[]</div> <div></div> </div>	
1	{ABC,DEF,GHI,JKL}

c1に'ABC'を持つ行を指定して、その要素の2番目を'ZZZ'に更新する。

Query Editor	Query History
<pre>1 update array_table set c1[2] = 'ZZZ' where 'ABC' = any(c1); 2 3 select * from array_table where 'ABC' = any(c1); </pre>	
Data Output	Explain Messages Notifications
<div> <div>c1</div> <div>text[]</div> <div></div> </div>	
1	{ABC,ZZZ,GHI,JKL}

8-3-9 NULL

Query EditorQuery History

1drop table tbl;
2
3create table tbl(c1 integer, c2 text);
4
5insert into tbl(c1) values (1);
6insert into tbl(c1, c2) values (2, 'BBB');
7insert into tbl(c1) values (3);
8
9select * from tbl;

Data OutputExplainMessagesNotifications

	c1 integer	c2 text
1		[null]
2		BBB
3		[null]

Query EditorQuery History

1select * from tbl where c2 is null;

Data OutputExplainMessagesNotifications

	c1 integer	c2 text
1		[null]
2		[null]

Query EditorQuery History

1select * from tbl where c2 is not null;

Data OutputExplainMessagesNotifications

	c1 integer	c2 text
1		BBB

並べ替えでNULLを最初もしくは最後にするかを設定できる。

まずは通常の並べ替えを行う。

Query EditorQuery History

1select * from tbl order by c2;

Data OutputExplainMessagesNotifications

	c1 integer	c2 text
1		BBB
2		[null]
3		[null]

次に、NULLを含んだ並べ替えをする。

Query Editor

Query History

1 select * from tbl order by c2 nulls first;

Data Output

Explain

Messages

Notifications

	c1 integer	c2 text
1	1	[null]
2	3	[null]
3	2	BBB

8-3-10 キャスト

データ型の変更ができる。

Query Editor	Query History				
<pre>1 select cast ('123' as int);</pre>					
Data Output	Explain Messages Notifications				
<table> <tr> <th></th><th>int4 integer</th></tr> <tr> <td>1</td><td>123</td></tr> </table>		int4 integer	1	123	
	int4 integer				
1	123				

Query Editor	Query History				
<pre>1 select '123'::int;</pre>					
Data Output	Explain Messages Notifications				
<table> <tr> <th></th><th>int4 integer</th></tr> <tr> <td>1</td><td>123</td></tr> </table>		int4 integer	1	123	
	int4 integer				
1	123				

Query Editor

Query History

1

drop table tbl;

2

3

create table tbl(c1 integer, c2 integer);

4

insert into tbl values(123, 456);

5

6

select * from tbl;

Data Output

Explain

Messages

Notifications

c1

integer

c2

integer

1

123

456

integer型のc1, c2の値をtext型にして、「-」で連結したものを、selectで抽出する。

Query Editor	Query History				
<pre>1 select c1::text '-' c2::text from tbl limit 1;</pre>					
Data Output	Explain Messages Notifications				
<table> <tr> <th></th><th>?column? text</th></tr> <tr> <td>1</td><td>123-456</td></tr> </table>		?column? text	1	123-456	
	?column? text				
1	123-456				

