**Table**

public.cp\_day

(

"time" timestamp without time zone NOT NULL,

open double precision,

high double precision,

low double precision,

close double precision,

volume integer,

symbol character varying COLLATE pg\_catalog."default" NOT NULL,

CONSTRAINT cp\_day\_pkey PRIMARY KEY ("time", symbol)

)

public.cp\_min

(

"time" timestamp without time zone NOT NULL,

open double precision,

high double precision,

low double precision,

close double precision,

volume integer,

symbol text COLLATE pg\_catalog."default" NOT NULL,

CONSTRAINT cp\_min\_pkey PRIMARY KEY ("time", symbol)

)

public.danh\_sach\_ma\_ck

(

symbol text COLLATE pg\_catalog."default",

id integer,

type text COLLATE pg\_catalog."default",

exchange text COLLATE pg\_catalog."default",

en\_organ\_name text COLLATE pg\_catalog."default",

en\_organ\_short\_name text COLLATE pg\_catalog."default",

organ\_short\_name text COLLATE pg\_catalog."default",

organ\_name text COLLATE pg\_catalog."default"

)

public.hop\_dong\_tuong\_lai

(

symbol text COLLATE pg\_catalog."default"

)

public.ma\_ck\_hnx

(

symbol text COLLATE pg\_catalog."default"

)

public.ma\_ck\_hose

(

symbol text COLLATE pg\_catalog."default"

)

public.ma\_ck\_upcom

(

symbol text COLLATE pg\_catalog."default"

)

public.ma\_index

(

symbol text COLLATE pg\_catalog."default"

)

public.ma\_vn30

(

symbol text COLLATE pg\_catalog."default"

)

public.ma\_hnx30

(

symbol text COLLATE pg\_catalog."default"

)

public.stock\_price\_board

(

symbol text COLLATE pg\_catalog."default", (mã)

ceiling integer, (trần)

floor integer, (sàn)

ref\_price integer, (TC)

stock\_type text COLLATE pg\_catalog."default", (loại cổ phiếu)

exchange text COLLATE pg\_catalog."default", (hệ thống giao dịch)

last\_trading\_date text COLLATE pg\_catalog."default", (ngày giao dịch cuối)

listed\_share integer,

type text COLLATE pg\_catalog."default",

id double precision,

organ\_name text COLLATE pg\_catalog."default",

prior\_close\_price double precision,

match\_price integer, (Giá)

match\_vol integer, (KL)

accumulated\_volume integer, (Tổng khối lượng)

accumulated\_value double precision, (Tổng giá trị)

avg\_match\_price double precision,

highest integer, (cao)

lowest integer, (Giá sàn)

match\_type text COLLATE pg\_catalog."default",

foreign\_sell\_volume integer, (nước ngoài bán)

foreign\_buy\_volume integer, (nước ngoài mua)

current\_room integer,

total\_room double precision, (số lượng CPLH)

bid\_1\_price double precision,

bid\_1\_volume double precision,

bid\_2\_price double precision,

bid\_2\_volume double precision,

bid\_3\_price double precision,

bid\_3\_volume double precision,

ask\_1\_price double precision,

ask\_1\_volume double precision,

ask\_2\_price double precision,

ask\_2\_volume double precision,

ask\_3\_price double precision,

ask\_3\_volume double precision

)

public.chi\_so\_tai\_chinh

(

symbol text COLLATE pg\_catalog."default" NOT NULL,

nam double precision NOT NULL,

ky double precision NOT NULL,

vay\_nh\_dh\_vcsh double precision,

no\_vcsh double precision,

tsc\_d\_von\_csh double precision,

von\_csh\_von\_dieu\_le double precision,

vong\_quay\_tai\_san double precision,

vong\_quay\_tsc\_d double precision,

so\_ngay\_thu\_tien\_binh\_quan double precision,

so\_ngay\_ton\_kho\_binh\_quan double precision,

so\_ngay\_thanh\_toan\_binh\_quan double precision,

chu\_ky\_tien double precision,

vong\_quay\_hang\_ton\_kho double precision,

bien\_ebit double precision,

bien\_loi\_nhuan\_gop double precision,

bien\_loi\_nhuan\_rong double precision,

roe double precision,

roic double precision,

roa double precision,

ebitda double precision,

ebit double precision,

ty\_suat\_co\_tuc double precision,

chi\_so\_thanh\_toan\_hien\_tai double precision,

chi\_so\_thanh\_toan\_tien\_mat double precision,

chi\_so\_thanh\_toan\_nhanh double precision,

kha\_nang\_chi\_tra\_lai\_vay double precision,

don\_bay\_tai\_chinh double precision,

pb double precision,

von\_hoa double precision,

so\_cp\_luu\_hanh double precision,

pe double precision,

ps double precision,

p\_cash\_flow double precision,

eps double precision,

bvps double precision,

ev\_ebitda double precision,

CONSTRAINT chi\_so\_tai\_chinh\_pkey PRIMARY KEY (cp, nam, ky)

)

CREATE TABLE IF NOT EXISTS public.intraday

(

"time" timestamp without time zone,

price double precision,

volume integer,

match\_type text COLLATE pg\_catalog."default",

symbol text COLLATE pg\_catalog."default",

id integer NOT NULL,

CONSTRAINT intraday\_pkey PRIMARY KEY (id)

)

**VIEW:**

CREATE OR REPLACE VIEW public.symbol\_price\_changes

AS

WITH ranked\_prices AS (

SELECT cp\_day.symbol,

cp\_day.close AS latest\_price,

lag(cp\_day.close) OVER (PARTITION BY cp\_day.symbol ORDER BY cp\_day."time") AS ref\_price,

row\_number() OVER (PARTITION BY cp\_day.symbol ORDER BY cp\_day."time" DESC) AS rn

FROM cp\_day

)

SELECT spb.symbol,

dsm.exchange,

dsm.organ\_name,

spb.ref\_price,

spb.match\_price AS latest\_close,

spb.match\_price - spb.ref\_price AS change\_value,

(spb.match\_price - spb.ref\_price) / spb.ref\_price \* 100::double precision AS change\_percentage

FROM stock\_price\_board spb

JOIN danh\_sach\_ma\_ck dsm ON spb.symbol = dsm.symbol

UNION ALL

SELECT mi.symbol,

NULL::text AS exchange,

NULL::text AS organ\_name,

rp.ref\_price,

rp.latest\_price AS latest\_close,

rp.latest\_price - rp.ref\_price AS change\_value,

(rp.latest\_price - rp.ref\_price) / NULLIF(rp.ref\_price, 0::double precision) \* 100::double precision AS change\_percentage

FROM ma\_index mi

JOIN ranked\_prices rp ON mi.symbol = rp.symbol::text

WHERE rp.rn <= 1;

CREATE OR REPLACE VIEW public.co\_ban

AS

SELECT ds.symbol,

dsm.exchange,

cs.von\_hoa AS thi\_gia\_von\_ty,

cs.eps,

cs.pe,

cs.pb,

cs.ps,

cs.roa,

cs.roe

FROM ( SELECT ma\_ck\_hose.symbol

FROM ma\_ck\_hose

UNION

SELECT ma\_ck\_hnx.symbol

FROM ma\_ck\_hnx

UNION

SELECT ma\_ck\_upcom.symbol

FROM ma\_ck\_upcom) ds

JOIN danh\_sach\_ma\_ck dsm ON ds.symbol = dsm.symbol

JOIN chi\_so\_tai\_chinh cs ON ds.symbol = cs.symbol

WHERE cs.nam = (( SELECT max(chi\_so\_tai\_chinh.nam) AS max

FROM chi\_so\_tai\_chinh

WHERE chi\_so\_tai\_chinh.symbol = ds.symbol)) AND cs.ky = (( SELECT max(chi\_so\_tai\_chinh.ky) AS max

FROM chi\_so\_tai\_chinh

WHERE chi\_so\_tai\_chinh.symbol = ds.symbol AND chi\_so\_tai\_chinh.nam = cs.nam));

CREATE OR REPLACE VIEW public.ma\_ck\_niemyet\_all

AS

SELECT ma\_ck\_hose.symbol

FROM ma\_ck\_hose

UNION

SELECT ma\_ck\_hnx.symbol

FROM ma\_ck\_hnx

UNION

SELECT ma\_ck\_upcom.symbol

FROM ma\_ck\_upcom

UNION

SELECT hop\_dong\_tuong\_lai.symbol

FROM hop\_dong\_tuong\_lai

UNION

SELECT ma\_index.symbol

FROM ma\_index;

CREATE OR REPLACE VIEW public.tong\_hop

AS

SELECT spb.symbol,

spb.ref\_price AS "Tham chiếu",

cd.open AS "Mở cửa",

cd.low AS "Thấp",

cd.high AS "Cao",

spb.accumulated\_volume AS "Khối lượng",

spb.accumulated\_value AS "Giá trị",

( SELECT avg(last\_10\_days.volume) AS avg

FROM ( SELECT cp\_day.volume

FROM cp\_day

WHERE cp\_day.symbol::text = spb.symbol

ORDER BY cp\_day."time" DESC

LIMIT 10) last\_10\_days) AS "KLTB 10 ngày",

cst.von\_hoa AS "Thị giá vốn tỷ",

cst.so\_cp\_luu\_hanh AS "Số lượng CPLH",

cst.pe AS "P/E",

cst.eps AS "EPS"

FROM stock\_price\_board spb

JOIN cp\_day cd ON spb.symbol = cd.symbol::text

JOIN chi\_so\_tai\_chinh cst ON spb.symbol = cst.symbol

WHERE cd."time" = (( SELECT max(cp\_day."time") AS max

FROM cp\_day

WHERE cp\_day.symbol::text = spb.symbol)) AND cst.nam = (( SELECT max(chi\_so\_tai\_chinh.nam) AS max

FROM chi\_so\_tai\_chinh

WHERE chi\_so\_tai\_chinh.symbol = spb.symbol)) AND cst.ky = (( SELECT max(chi\_so\_tai\_chinh.ky) AS max

FROM chi\_so\_tai\_chinh

WHERE chi\_so\_tai\_chinh.symbol = spb.symbol AND chi\_so\_tai\_chinh.nam = cst.nam));

CREATE OR REPLACE VIEW public.index\_info

AS

WITH price\_changes AS (

SELECT stock\_price\_board.symbol,

CASE

WHEN stock\_price\_board.match\_price > stock\_price\_board.ref\_price THEN 'increase'::text

WHEN stock\_price\_board.match\_price < stock\_price\_board.ref\_price THEN 'decrease'::text

ELSE 'stable'::text

END AS change\_type

FROM stock\_price\_board

)

SELECT 'VNINDEX'::text AS index,

sum(th."Khối lượng") AS "KL tổng",

sum(th."Giá trị") AS "GT tổng",

count(

CASE

WHEN pc.change\_type = 'decrease'::text THEN 1

ELSE NULL::integer

END) AS decrease,

count(

CASE

WHEN pc.change\_type = 'stable'::text THEN 1

ELSE NULL::integer

END) AS stable,

count(

CASE

WHEN pc.change\_type = 'increase'::text THEN 1

ELSE NULL::integer

END) AS increase

FROM tong\_hop th

JOIN price\_changes pc ON th.symbol = pc.symbol

JOIN ma\_ck\_hose mch ON th.symbol = mch.symbol

UNION ALL

SELECT 'HNXINDEX'::text AS index,

sum(th."Khối lượng") AS "KL tổng",

sum(th."Giá trị") AS "GT tổng",

count(

CASE

WHEN pc.change\_type = 'decrease'::text THEN 1

ELSE NULL::integer

END) AS decrease,

count(

CASE

WHEN pc.change\_type = 'stable'::text THEN 1

ELSE NULL::integer

END) AS stable,

count(

CASE

WHEN pc.change\_type = 'increase'::text THEN 1

ELSE NULL::integer

END) AS increase

FROM tong\_hop th

JOIN price\_changes pc ON th.symbol = pc.symbol

JOIN ma\_ck\_hnx mch ON th.symbol = mch.symbol

UNION ALL

SELECT 'UPCOMINDEX'::text AS index,

sum(th."Khối lượng") AS "KL tổng",

sum(th."Giá trị") AS "GT tổng",

count(

CASE

WHEN pc.change\_type = 'decrease'::text THEN 1

ELSE NULL::integer

END) AS decrease,

count(

CASE

WHEN pc.change\_type = 'stable'::text THEN 1

ELSE NULL::integer

END) AS stable,

count(

CASE

WHEN pc.change\_type = 'increase'::text THEN 1

ELSE NULL::integer

END) AS increase

FROM tong\_hop th

JOIN price\_changes pc ON th.symbol = pc.symbol

JOIN ma\_ck\_upcom mcu ON th.symbol = mcu.symbol

UNION ALL

SELECT 'VN30'::text AS index,

sum(th."Khối lượng") AS "KL tổng",

sum(th."Giá trị") AS "GT tổng",

count(

CASE

WHEN pc.change\_type = 'decrease'::text THEN 1

ELSE NULL::integer

END) AS decrease,

count(

CASE

WHEN pc.change\_type = 'stable'::text THEN 1

ELSE NULL::integer

END) AS stable,

count(

CASE

WHEN pc.change\_type = 'increase'::text THEN 1

ELSE NULL::integer

END) AS increase

FROM tong\_hop th

JOIN price\_changes pc ON th.symbol = pc.symbol

JOIN ma\_vn30 mv ON th.symbol = mv.symbol

UNION ALL

SELECT 'HNX30'::text AS index,

sum(th."Khối lượng") AS "KL tổng",

sum(th."Giá trị") AS "GT tổng",

count(

CASE

WHEN pc.change\_type = 'decrease'::text THEN 1

ELSE NULL::integer

END) AS decrease,

count(

CASE

WHEN pc.change\_type = 'stable'::text THEN 1

ELSE NULL::integer

END) AS stable,

count(

CASE

WHEN pc.change\_type = 'increase'::text THEN 1

ELSE NULL::integer

END) AS increase

FROM tong\_hop th

JOIN price\_changes pc ON th.symbol = pc.symbol

JOIN ma\_hnx30 mh ON th.symbol = mh.symbol;