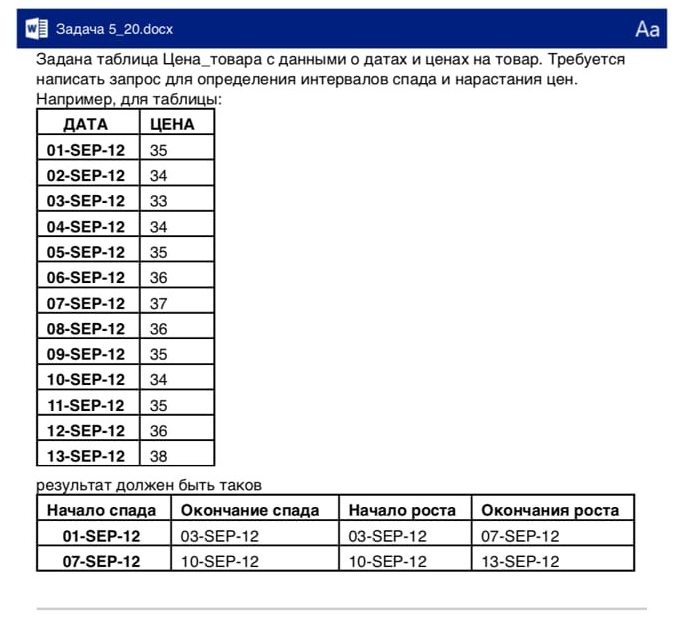
Ситдиков Филипп, гр № 3530203/70102

**Дополнительное задание 5.20**



**Решение**

WITH smain AS (

SELECT

дата,

цена,

ROWNUM AS lvl

FROM

(

SELECT

дата,

цена

FROM

prices

ORDER BY

дата

)

),sa AS (

SELECT

CASE

WHEN ( lvl IN (

SELECT

MAX(lvl)

FROM

smain

) ) THEN NULL

ELSE дата

END

AS "Начало спада",

ROWNUM AS rwn

FROM

smain outer

WHERE

(

(

lvl = 1

AND цена > (

SELECT

цена

FROM

smain

WHERE

lvl = outer.lvl + 1

)

)

OR (

цена > (

SELECT

цена

FROM

smain

WHERE

lvl = outer.lvl - 1

)

AND цена > (

SELECT

цена

FROM

smain

WHERE

lvl = outer.lvl + 1

)

)

OR (

цена = (

SELECT

цена

FROM

smain

WHERE

lvl = outer.lvl - 1

)

AND цена > (

SELECT

цена

FROM

smain

WHERE

lvl = outer.lvl + 1

)

)

OR lvl IN (

SELECT

MAX(lvl)

FROM

smain

)

)

),sb AS (

SELECT

CASE

WHEN ( lvl IN (

SELECT

MIN(lvl)

FROM

smain

) ) THEN NULL

ELSE дата

END

AS "Окончание спада",

ROWNUM AS rwn

FROM

smain outer

WHERE

(

(

цена < (

SELECT

цена

FROM

smain

WHERE

lvl = outer.lvl - 1

)

AND lvl IN (

SELECT

MAX(lvl)

FROM

smain

)

)

OR (

цена < (

SELECT

цена

FROM

smain

WHERE

lvl = outer.lvl - 1

)

AND цена < (

SELECT

цена

FROM

smain

WHERE

lvl = outer.lvl + 1

)

)

OR (

цена < (

SELECT

цена

FROM

smain

WHERE

lvl = outer.lvl - 1

)

AND цена = (

SELECT

цена

FROM

smain

WHERE

lvl = outer.lvl + 1

)

)

OR lvl IN (

SELECT

MIN(lvl)

FROM

smain

)

)

),sc AS (

SELECT

CASE

WHEN ( lvl IN (

SELECT

MAX(lvl)

FROM

smain

) ) THEN NULL

ELSE дата

END

AS "Начало роста",

ROWNUM AS rwn

FROM

smain outer

WHERE

(

(

lvl = 1

AND цена < (

SELECT

цена

FROM

smain

WHERE

lvl = outer.lvl + 1

)

)

OR (

цена < (

SELECT

цена

FROM

smain

WHERE

lvl = outer.lvl - 1

)

AND цена < (

SELECT

цена

FROM

smain

WHERE

lvl = outer.lvl + 1

)

)

OR (

цена = (

SELECT

цена

FROM

smain

WHERE

lvl = outer.lvl - 1

)

AND цена < (

SELECT

цена

FROM

smain

WHERE

lvl = outer.lvl + 1

)

)

OR lvl IN (

SELECT

MAX(lvl)

FROM

smain

)

)

),sd AS (

SELECT

CASE

WHEN ( lvl IN (

SELECT

MIN(lvl)

FROM

smain

) ) THEN NULL

ELSE дата

END

AS "Окончание роста",

ROWNUM AS rwn

FROM

smain outer

WHERE

(

(

цена > (

SELECT

цена

FROM

smain

WHERE

lvl = outer.lvl - 1

)

AND lvl IN (

SELECT

MAX(lvl)

FROM

smain

)

)

OR (

цена > (

SELECT

цена

FROM

smain

WHERE

lvl = outer.lvl - 1

)

AND цена > (

SELECT

цена

FROM

smain

WHERE

lvl = outer.lvl + 1

)

)

OR (

цена > (

SELECT

цена

FROM

smain

WHERE

lvl = outer.lvl - 1

)

AND цена = (

SELECT

цена

FROM

smain

WHERE

lvl = outer.lvl + 1

)

)

OR lvl IN (

SELECT

MIN(lvl)

FROM

smain

)

)

) SELECT

sa."Начало спада",

sb."Окончание спада",

sc."Начало роста",

sd."Окончание роста"

FROM

sa,

sb,

sc,

sd

WHERE

(

sa.rwn = ( sb.rwn - 1 )

AND ( sb.rwn - 1 ) = sc.rwn

AND sc.rwn = ( sd.rwn - 1 )

)

OR (

sa.rwn = ( sb.rwn - 1 )

AND ( sb.rwn - 1 ) = sc.rwn

AND sd.rwn = 1

AND sc."Начало роста" IS NULL

)

OR (

sc.rwn = ( sd.rwn - 1 )

AND ( sd.rwn - 1 ) = sa.rwn

AND sb.rwn = 1

AND sa."Начало спада" IS NULL

);

**Разбор запроса**

* подзапрос smain возвращает упорядоченную по датам таблицу prices

WITH smain AS (

SELECT

дата,

цена,

ROWNUM AS lvl

FROM

(

SELECT

дата,

цена

FROM

prices

ORDER BY

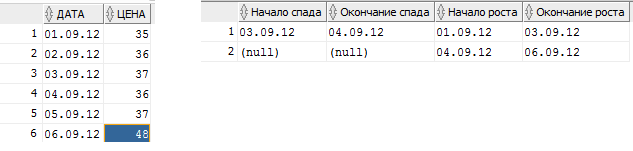
дата

)

)

* подзапрос **sa** возвращает столбец с датами начала спада, которые определяются **c помощью корелированных подзапросов**:
* запись первая и на следующий день цена понизилась
* цена сегодня больше, чем вчера и завтра
* цена вчера была такой же и завтра повысилась

также возвращается последняя запись (так как последняя запись не может быть началом спада) - max(lvl), которая превращается в null для того чтобы вывести null, если промежутков роста больше, чем промежутков спада, как в примере:



sa AS (

SELECT

CASE

WHEN ( lvl IN (

SELECT

MAX(lvl)

FROM

smain

) ) THEN NULL

ELSE дата

END

AS "Начало спада",

ROWNUM AS rwn

FROM

smain outer

WHERE

(

(

lvl = 1

AND цена > (

SELECT

цена

FROM

smain

WHERE

lvl = outer.lvl + 1

)

)

OR (

цена > (

SELECT

цена

FROM

smain

WHERE

lvl = outer.lvl - 1

)

AND цена > (

SELECT

цена

FROM

smain

WHERE

lvl = outer.lvl + 1

)

)

OR (

цена = (

SELECT

цена

FROM

smain

WHERE

lvl = outer.lvl - 1

)

AND цена > (

SELECT

цена

FROM

smain

WHERE

lvl = outer.lvl + 1

)

)

OR lvl IN (

SELECT

MAX(lvl)

FROM

smain

)

)

)

* sb, sc и sd выполнены аналогично
* **Основной запрос**, который соединяет подзапросы **sa, sb, sc и sd** по rowid, а так как в sb и sd null-строка первая, приходится добавлять доп условия

SELECT

sa."Начало спада",

sb."Окончание спада",

sc."Начало роста",

sd."Окончание роста"

FROM

sa,

sb,

sc,

sd

WHERE

(

sa.rwn = ( sb.rwn - 1 )

AND ( sb.rwn - 1 ) = sc.rwn

AND sc.rwn = ( sd.rwn - 1 )

)

OR (

sa.rwn = ( sb.rwn - 1 )

AND ( sb.rwn - 1 ) = sc.rwn

AND sd.rwn = 1

AND sc."Начало роста" IS NULL

)

OR (

sc.rwn = ( sd.rwn - 1 )

AND ( sd.rwn - 1 ) = sa.rwn

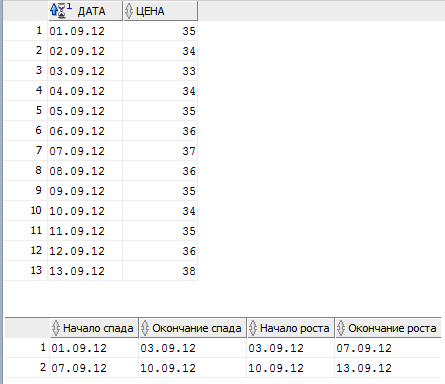
AND sb.rwn = 1

AND sa."Начало спада" IS NULL

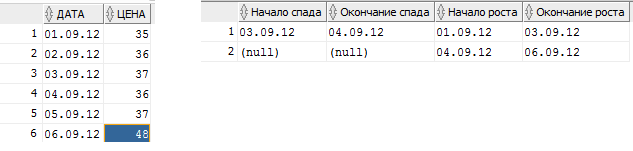
);

**Примеры**

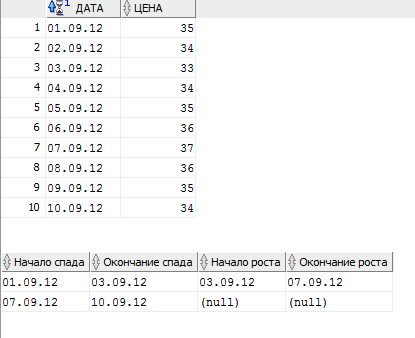
1. Пример из самого задания:



1. Промежутков роста больше, чем спада, добавляются null-ы



1. Промежутков спада больше, чем роста



1. Один спад, один рост

