```sql
select distinct F.customerid , F.recency, F.frequency, F.monetary, F.r_score, F.fm_score,

Case

when F.r_score >= 5 and F.fm_score >= 5

or   F.r_score >= 5 and F.fm_score =4

or   F.r_score = 4  and F.fm_score >= 5 then 'Champions'


when F.r_score >= 5 and F.fm_score = 2

or   F.r_score = 4 and F.fm_score = 2

or   F.r_score = 3 and F.fm_score = 3

or   F.r_score = 4  and F.fm_score >= 3 then 'Potential Loyalists'


when F.r_score >= 5 and F.fm_score = 3

or   F.r_score = 4 and F.fm_score = 4

or   F.r_score = 3 and F.fm_score >= 5

or   F.r_score = 3  and F.fm_score >= 4 then 'Loyal Customers'


when F.r_score >= 5 and F.fm_score = 1 then 'Recent Customers'


when F.r_score = 4 and F.fm_score = 1

or   F.r_score = 3  and F.fm_score = 1 then 'Promising'


when F.r_score = 3 and F.fm_score = 2

or   F.r_score = 2 and F.fm_score = 3

or   F.r_score = 2  and F.fm_score = 2 then 'Customers Needing Attention'


when F.r_score = 2 and F.fm_score >= 5

or   F.r_score = 2 and F.fm_score = 4

or   F.r_score = 1  and F.fm_score = 3 then 'At Risk'
```

when F.r_score = 1 and F.fm_score >= 5

or   F.r_score = 1 and F.fm_score = 4 then 'Cant Lose Them'

when F.r_score = 1 and F.fm_score = 2

or   F.r_score = 2 and F.fm_score = 1 then 'Hibernating'

when F.r_score = 1 and F.fm_score <= 1 then 'Lost'

End cust_segment

from (select distinct L.customerid , L.recency, L.frequency, L.monetary,

NTILE(5) over(order by L.recency desc) as r_score ,

Avg((L.frequency+L.monetary)/2) over(rows between current row and current row) ::numeric(10) as fm_score

from (

with freq As (

select customerid , count(distinct invoiceno) as frequency

from online_retail

group by customerid

having customerid != ''),

recency As (select distinct X.customerid,

ceil(first_value((extract(day from (Y.reference_date - TO_TIMESTAMP(invoicedate,'mm-dd-yyyy HH24:MI')))*24

                                 + extract(hour from (Y.reference_date - TO_TIMESTAMP(invoicedate,'mm-dd-yyyy HH24:MI'))))/24)

over(partition by X.customerid order by X.invoicedate desc)) as recency

from (select max(TO_TIMESTAMP(invoicedate,'mm-dd-yyyy HH24:MI')) as reference_date

from online_retail where customerid != '') Y , online_retail X

where customerid != '')

```sql
select  distinct O.customerid ,  freq.frequency , recency.recency

, sum(O.unitprice) over(partition by O.customerid) as monetary

from online_retail O join freq using(customerid)

join recency using(customerid)

where O.customerid != ''            ) As L) F

where F.customerid in ('13256' ,'15118', '12875','13366','17752', '14609','15753','16881')
```