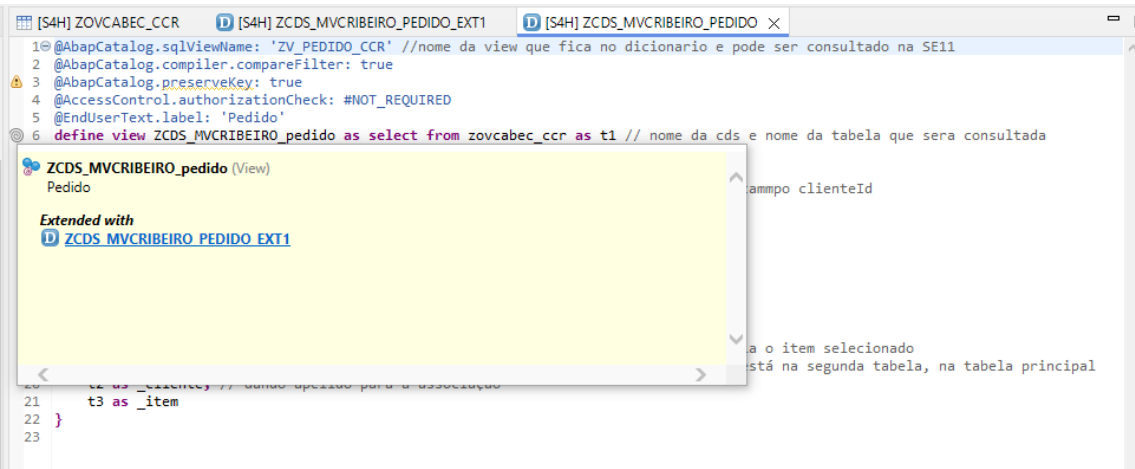


Adicionando mais um campo na CDS sem mexer na original.  
Aqui foi feito através de **EXTENSÃO**.

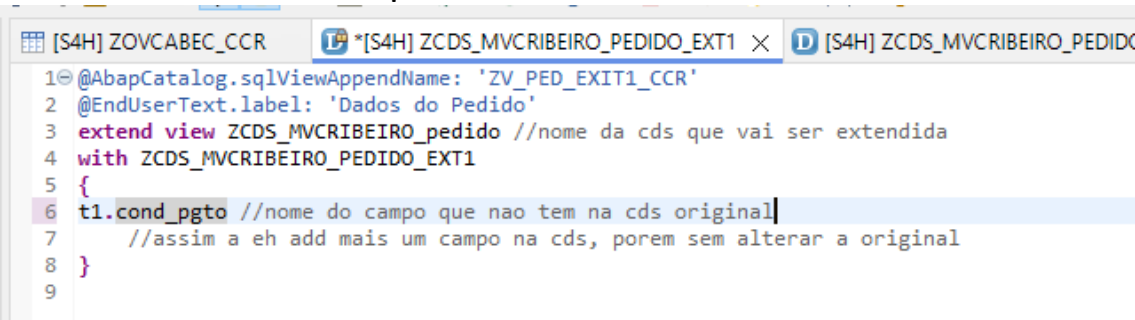


The screenshot shows the SAP IDE editor with the following SQL code:

```
1 @AbapCatalog.sqlViewName: 'ZV_PEDIDO_CCR' //nome da view que fica no dicionário e pode ser consultado na SE11
2 @AbapCatalog.compiler.compareFilter: true
3 @AbapCatalog.preserveKey: true
4 @AccessControl.authorizationCheck: #NOT_REQUIRED
5 @EndUserText.label: 'Pedido'
6 define view ZCDS_MVCRIBEIRO_pedido as select from zovcabec_ccr as t1 // nome da cds e nome da tabela que sera consultada
```

A tooltip for the view `ZCDS_MVCRIBEIRO_pedido` is displayed, indicating it is extended from `ZCDS_MVCRIBEIRO_PEDIDO_EXT1`.

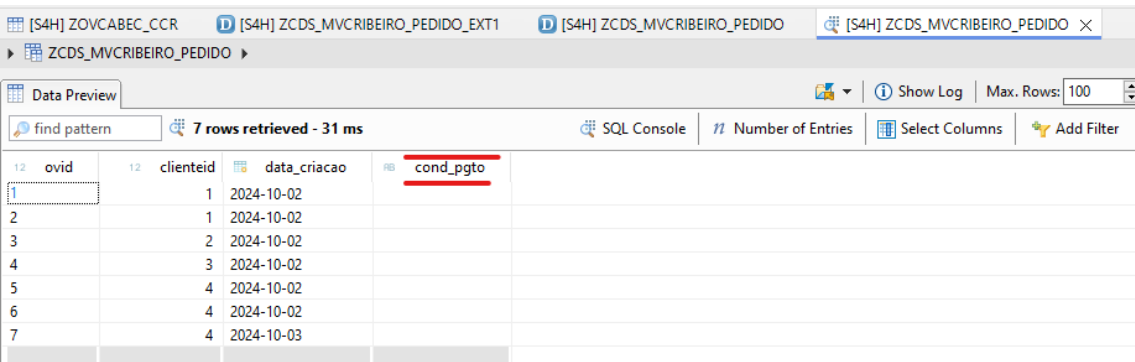
Cria-se outra CDS do tipo **EXTEND VIEW**.



The screenshot shows the SAP IDE editor with the following SQL code:

```
1 @AbapCatalog.sqlViewAppendName: 'ZV_PED_EXT1_CCR'
2 @EndUserText.label: 'Dados do Pedido'
3 extend view ZCDS_MVCRIBEIRO_pedido //nome da cds que vai ser extendida
4 with ZCDS_MVCRIBEIRO_PEDIDO_EXT1
5 {
6 t1.cond_pgto //nome do campo que nao tem na cds original
7 //assim a eh add mais um campo na cds, porem sem alterar a original
8 }
9
```

RESULTADO



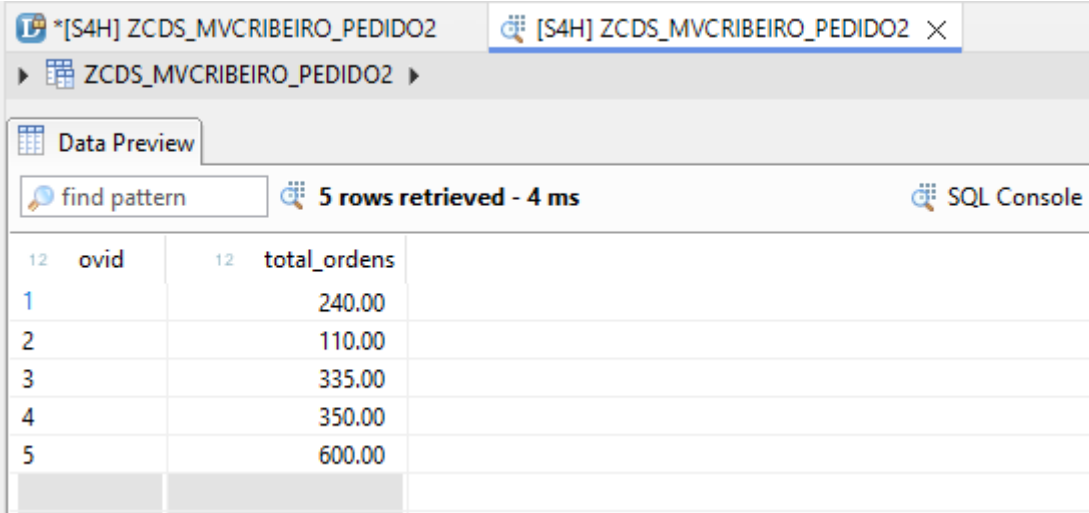
The screenshot shows the SAP IDE Data Preview window with the following data:

oivid	clienteid	data_criacao	cond_pgto
1	1	2024-10-02	
2	1	2024-10-02	
3	2	2024-10-02	
4	3	2024-10-02	
5	4	2024-10-02	
6	4	2024-10-02	
7	4	2024-10-03	

Abaixo estendendo uma cds através de **GROUP BY**  
Primeiro cria-se o group by.

```
1 @AbapCatalog.sqlViewName: 'ZV_PEDIDO2_CCR' //nome da view que fica no dicionario e pode ser consultado na SE11
2 @AbapCatalog.compiler.compareFilter: true
3 @AbapCatalog.preserveKey: true
4 @AccessControl.authorizationCheck: #NOT_REQUIRED
5 @EndUserText.label: 'pedido2'
6 define view ZCDS_MVCRIBEIRO_pedido2 as // nome da cds
7 select from zovcabec_ccr as t1 //nome da tabela que sera consultada
8
9 //juntando 2 tabelas atraves do campo ovid
10 inner join ZCDS_MVCRIBEIRO_ITEM_OV as t2 on t1.ovid = t2.Ovid
11 {
12     key t1.ovid,
13     sum( t2.PrecoTotal ) as total_ordens //somando todos os itens de acordo com ordemm de venda e add em total de ordens
14 }
15 group by t1.ovid // fazendo o agrupamento pela ordem de venda
16
```

## RESULTADO



ovid	total_ordens
1	240.00
2	110.00
3	335.00
4	350.00
5	600.00

Agora a extensão via **GROUP BY**  
Add uma anotação na view original

```
1 @AbapCatalog.sqlViewName: 'ZV_PEDIDO2_CCR' //nome da view que fica no dicionario e pode ser consultado na SE11
2 @AbapCatalog.compiler.compareFilter: true
3 @AbapCatalog.preserveKey: true
4 @AccessControl.authorizationCheck: #NOT_REQUIRED
5 @EndUserText.label: 'pedido2'
6
7 //precisa dessa anotação para fazer extensao com group by
8 @AbapCatalog.viewEnhancementCategory: [ #PROJECTION_LIST,#GROUP_BY ]
9
10
11 define view ZCDS_MVCRIBEIRO_pedido2 as // nome da cds
12 select from zovcabec_ccr as t1 //nome da tabela que sera consultada
13
14 //juntando 2 tabelas atraves do campo ovid
15 inner join ZCDS_MVCRIBEIRO_ITEM_OV as t2 on t1.ovid = t2.Ovid
16 {
17     key t1.ovid,
18     sum( t2.PrecoTotal ) as total_ordens //somando todos os itens de acordo com ordemm de venda e add em total de ordens
19 }
20 group by t1.ovid // fazendo o agrupamento pela ordem de venda
21
```

## Cria-se uma nova view do tipo **EXTEND VIEW**

```
1 @AbapCatalog.sqlViewAppendName: 'ZV_PED2_EXT1_CCR'
2 @EndUserText.label: 'Dados do Pedido'
3 extend view ZCDS_MVCRIBEIRO_pedido2 //nome da cds que vai ser extendida
4 with ZCDS_MVCRIBEIRO_PEDIDO2_EXT1
5 {
6     min(t2.PrecoTotal) as preco_minimo,
7     max(t2.PrecoTotal) as preco_maximo
8 }
9 group by t1.ovid
10
```

## RESULTADO

[S4H] ZCDS\_MVCRIBEIRO\_PEDIDO2

Data Preview

find pattern 5 rows retrieved - 49 ms SQL Console 11 Number

12	ovid	12	total_ordenSs	12	preco_minimo	12	preco_maximo
1			240.00		240.00		240.00
2			110.00		110.00		110.00
3			335.00		55.00		280.00
4			350.00		350.00		350.00
5			600.00		90.00		400.00


## AGORA ESTENDENDO ATRAVÉS DE **UNION**

```
3 @AbapCatalog.preserveKey: true
4 @AccessControl.authorizationCheck: #NOT_REQUIRED
5 @EndUserText.label: 'Parceiro'
6
7 //precisa dessa anotação para fazer extensao com UNION
8 @AbapCatalog.viewEnhancementCategory: [ #PROJECTION_LIST,#UNION ]
9
10 define view ZCDS_MVCRIBEIRO_PARCEIRO as // nome da cds
11
12 select from zcliente_ccr // nome da tabela que sera consultada
13 {
14     clienteid as Parceiroid,
15     nome as Nome
16     //como na segunda tabela tem a coluna email, foi criada esta coluna com o mesmo tipo
17     //cast('' as abap.char(255)) as Email
18 }
19
20
21 union // quando tem o mesmo dado nas 2 consultas, unio ã mostra no resultado essa repetição
22 //union all // quando tem o mesmo dado nas 2 consultas, union all mostra
23
24 select from zfornecedor_ccr // nome da tabela que sera consultada
25 {
26     fornecedor_id as Parceiroid,
27     nome as Nome
28     //email as Email
29 }
30
```

```
[S4H] ZCDS_MVCRIBEIRO_PARCEIRO [S4H] ZCDS_MVCRIBEIRO_PARCEIRO_EXT1 [S4H] ZCDS_MVCRIBEIRO_PARCEIRO
1 @AbapCatalog.sqlViewAppendName: 'ZV_PARC_EXT1_CCR'
2 @EndUserText.label: 'Dados Parceiro'
3
4 extend view ZCDS_MVCRIBEIRO_PARCEIRO //nome da cds que vai ser estendida
5 with ZCDS_MVCRIBEIRO_PARCEIRO_EXT1
6 {
7   cast('nao possui' as abap.char( 255 )) as Email
8 }
9
10 union
11 {
12   email
13 }
14 }
15 }
```

[S4H] ZCDS\_MVCRIBEIRO\_PARCEIRO [S4H] ZCDS\_MVCRIBEIRO\_PARCEIRO\_EXT1 [S4H] ZCDS\_MVCRIBEIRO\_PARCEIRO

► ZCDS\_MVCRIBEIRO\_PARCEIRO ►

Data Preview  Show Log

find pattern 7 rows retrieved - 4 ms SQL Console Number of Entries Select Colour

12	Parceiroid	Nome	Email
2	LOJA2	LOJA2@LOJA.COM.BR	
3	JOSE	JOSE@TESTE.COM.BR	
4	DIRCE	nao possui	
1	JOSE	nao possui	
2	PEDRO	nao possui	
3	JOAQUIM	nao possui	
1	LOJA1	LOJA1@LOJA.COM.BR	