

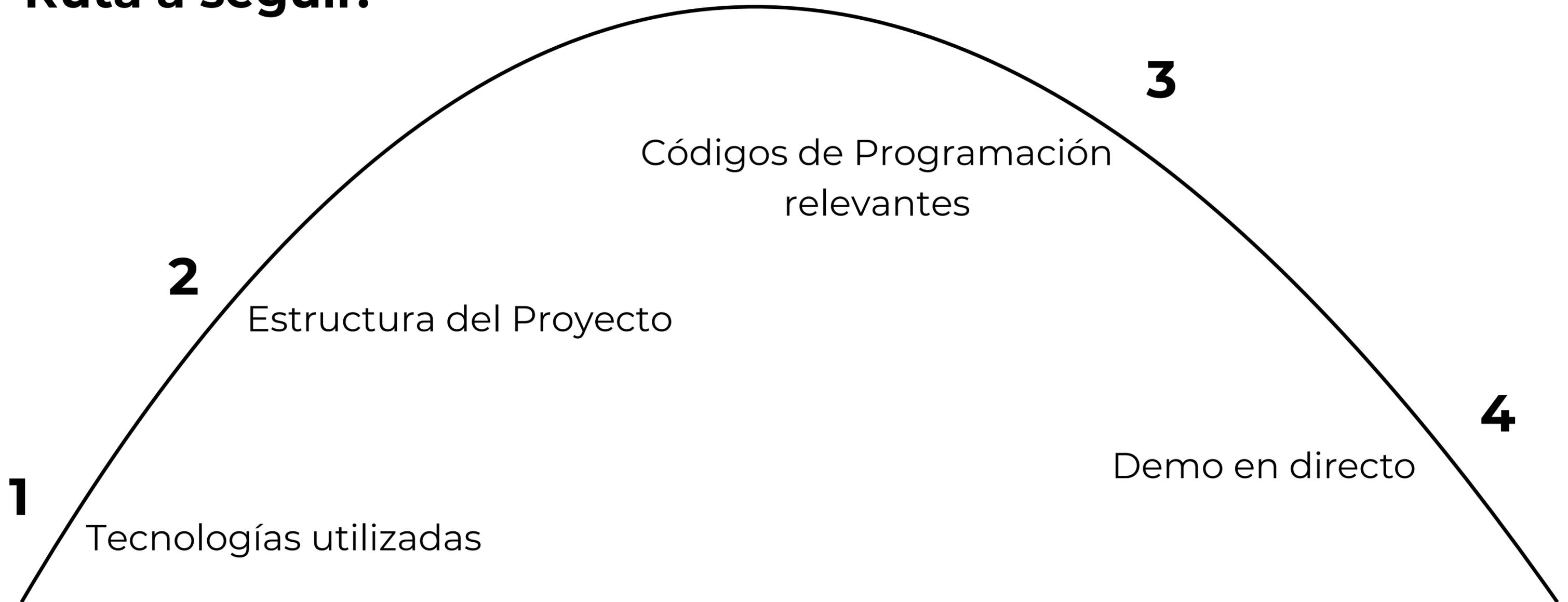
PROYECTO

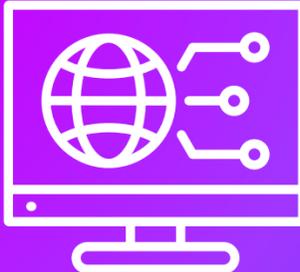
AID & TRAVEL

Por Alejandro Fernández Pérez



Ruta a seguir:





Tecnologías utilizadas

01

Lenguaje de programación REXX

02

Subsistema DB2

03

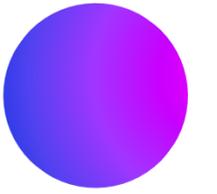
Uso de Datasets PDS y PDSE

04

Otras tecnologías como:

- ISPF.
- VS Code.

Lenguaje REXX



USO



Lógica de la demo

¿Por qué su elección?



Facilidad.
Sencillo.
Potente.
Interpretado.



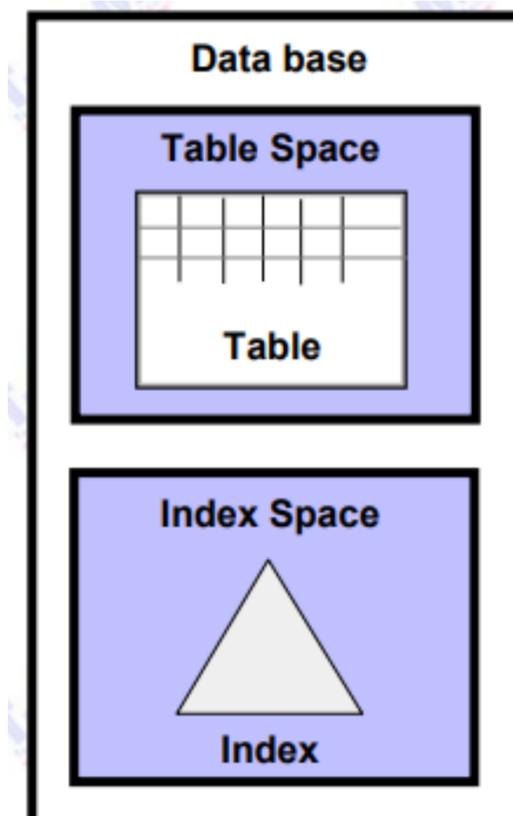
Subsistema DB2



USO



Almacenar datos o información



```
SELECT USUARIO, PSSWD, TIPO FROM TSOPSS1.USUARIOS
INSERT INTO TSOPSS1.USUARIOS (USUARIO, PSSWD, NOMBRE,
APELLIDOS, GENERO, TIPO, EDAD, PAIS) "
VALUES (?, ?, ?, ?, ?, ?, ?, ?)"
```

```
DELETE FROM TSOPSS1.USUARIOS
```

```
/* Conectar con la base de datos */
ADDRESS TSO "SUBCOM DSNREXX"
IF RC THEN S = RXSUBCOM('ADD','DSNREXX','DSNREXX')

/* Conectar a DB2 */
subsys = "DBDG"           ADDRESS DSNREXX "DISCONNECT"
user = userid()           ..
ADDRESS DSNREXX "CONNECT" subsys
```

Datasets PDS y PDSE

```
TS0PSS1.AANDT.CLIST(EJECUTOR)
```

```
ALloc fi(sysuexec) dataset('TS0PSS1.AANDT.REXX.EXEC') shr  
ALTLIB ACT user(EXEC)  
ALTLIB display
```

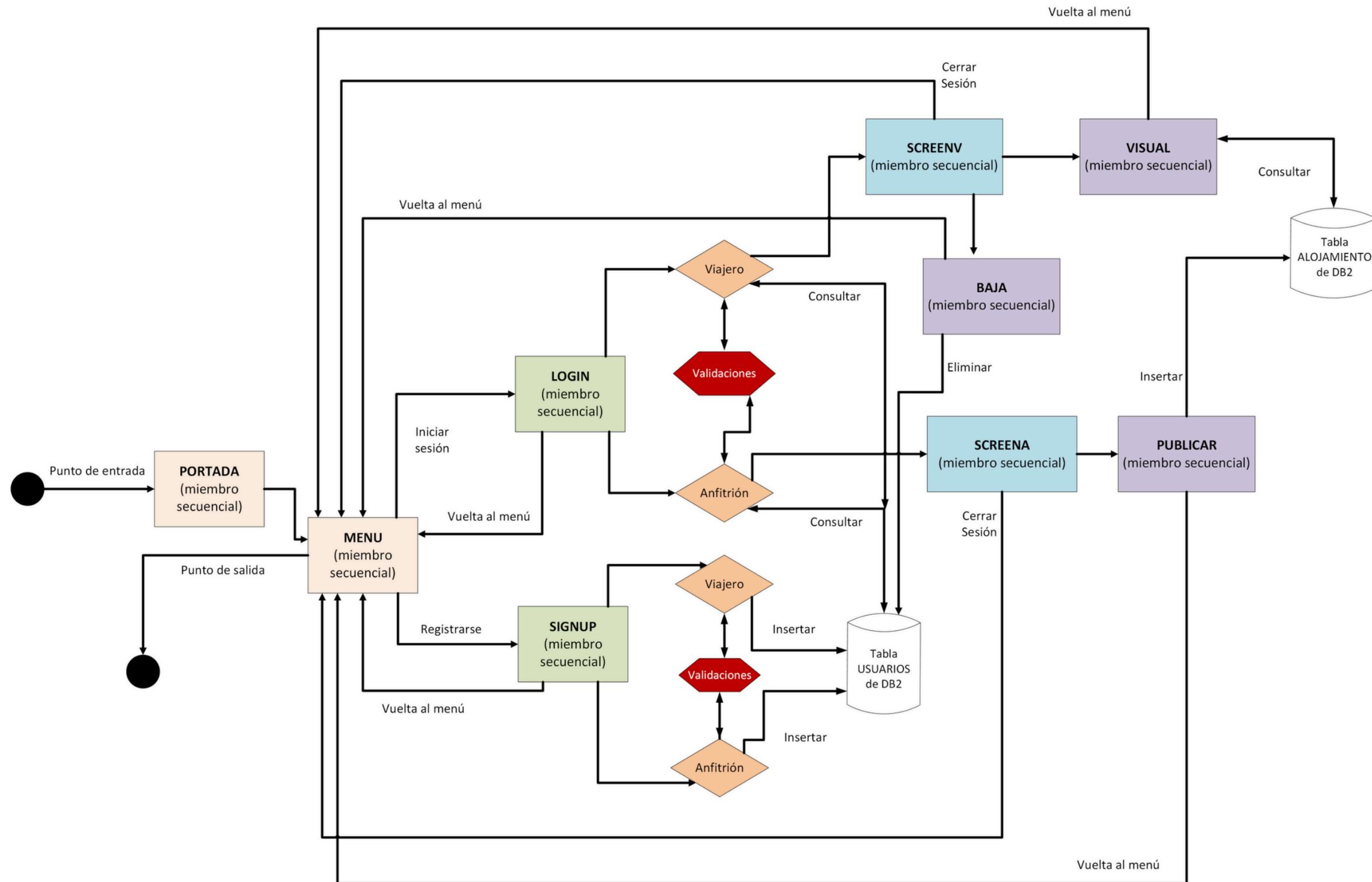
Pasos ejecución:

```
exec 'tsopss1.aandt.clist(ejecutor)'
```

portada 1º miembro y entrada a la demo

TS0PSS1.AANDT.REXX.EXEC			
Name	Prompt	Size	Cr
BAJA		66	202
LOGIN		182	202
MENU		56	202
PORTADA		31	202
PUBLICAR		256	202
SCREENA		132	202
SCREENV		142	202
SIGNUP		309	202
VISUAL		79	202

Descripción de la estructura



Códigos de Programación relevantes

La estructura de control REXX

SELECT

WHEN then do

end

OTHERWISE

END

PARSE PULL ENTRADA

SAY "Hola que tal?"

DO UNTIL

END

DO WHILE

END

DO FOREVER

END

Ejecución de otro miembro dentro
de un DS

```
address TSO "EXEC 'TSOPSS1.AANDT.REXX.EXEC(LOGIN)'"  
return
```

Códigos de Programación relevantes

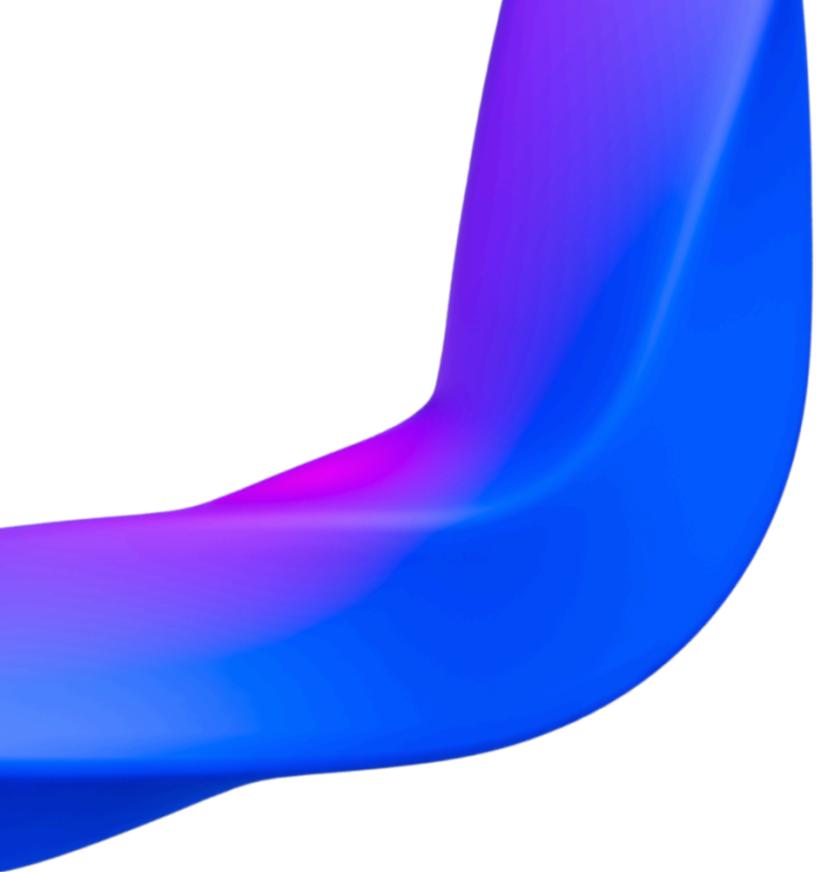
Variables globales

```
USERID = CHECKUSER  
ADDRESS ISPEXEC "VPUT (USERID) SHARED"
```

```
ADDRESS ISPEXEC "VGET (USERID) SHARED"  
USUARIO CX = USERID
```

La consulta SQL a base de datos en bucle

```
do while SQLCODE = 0  
  CAPTARSQL1 = "EXECSQL FETCH C1 INTO :NUM_PUNT, :SEDE_PUNT, :HOST_PUNT, "  
  CAPTARSQL2 = ":OFRECER_PUNT, :RECIBIR_PUNT, :UBICACION_PUNT"  
  CAPTARSQL = CAPTARSQL1 || CAPTARSQL2  
  ADDRESS DSMREXX CAPTARSQL  
  
  if SQLCODE = 0 then do  
say '-----'  
say '|'  
say '|                ALOJAMIENTO 'NUM_PUNT'. 'HOST_PUNT'' '|  
say '|                '|  
say '|                DESCRIPCION: 'SEDE_PUNT'' '|  
say '|                OFRECES: 'OFRECER_PUNT'' '|  
say '|                OBTIENES: 'RECIBIR_PUNT'' '|  
say '|                UBICACION: 'UBICACION_PUNT'' '|  
say '|                '|  
say '-----'  
say  
say  
say "Pulsa ENTER para ver el siguiente..."  
parse pull /* Espera a que el usuario pulse Enter */  
  end  
end
```

MUCHAS GRACIAS

¿Alguna pregunta?

