

- **Modelo de datos:** definir y crear la(s) tabla(s) que se considere(n) necesaria(s) teniendo como objetivo optimizar los tiempos de respuesta en la administración de la orden (creación de la orden, adición y retiro de productos del carrito, adición de los datos de factura y pago, cambio de estados de la orden).

1. Crear la rowkey :

nroOrden	idProducto	cantidad	IdCliente
6284	149	1	289
6284	148	2	289

data=ProductosOrden.csv
se agrego una columna con idCliente:
rowkey=nroO+idP+idC

rowkey

nroOrden	idCliente	fecha	estado
6284	289	6/1/2016	Finalizada

data= Orden.csv
Familia:columna 'Ord:est'

idProducto	idCategor	nombre	descripcion	noPiezas	pesoKg	precio	cantidadDisp
149	400	Jean Lev	non	1	7	188120	679,,,
148	700	Rastrillo Kyle	pede blandit	1	5	555251	698,,,

data=productos.csv
familia:columna 'Ord:nomP'

idClien	nombre	direccion	codCiud	email	nombreDeUsuar	contrase
289	Jaquelyn Cooke	Ap #7590 Pede. Rd.	7601	tincidunt.duit@vo.org	Lionel	nisi

data=cliente.csv
Familia:columna 'Def:nomC'

nroFactu	Fecha	noTarjeta	tipoTarje	noAprobaci	idEntid	nroOrd
3892	9/1/2016	2.70679E+11	DEBITO	505	4	6284

data=factura.csv
familia:Columna 'Def:nroF'

Tabla ('ARcard')

	Familia Ord							Familia Def							
rowkey	nroO	idP	cant	idC	fecO	est	nomC	nomP	precio	nroF	fecF	nroT	tipoT	nroA	idE
6284149289	6284	149	1	289	6/1/2016	Finalizada	Jaquelyn Cooke	Jean Lev	188120	3892	9/1/2016	270679299369	DEBITO	505	4

```
bd07@serverhadoop:~$ more load.py
import happybase
import csv

connection = happybase.Connection()
tables = connection.tables()

if "ARcard" in tables:
    connection.disable_table("ARcard")
    connection.delete_table("ARcard")

connection.create_table("ARcard", {"Ord" : dict(), "Def" : dict()})
table = connection.table("ARcard")

ifile = open("productosOrden4.csv", "rU")
reader = csv.reader(ifile, delimiter=",")
productosOrden = []
for row in reader:
    productosOrden.append(row)

ifile.close()

ifile = open("factura.csv", "rU")
reader = csv.reader(ifile, delimiter=",")
factura = []
for row in reader:
    factura.append(row)
ifile.close()

ifile = open("producto.csv", "rU")
reader = csv.reader(ifile, delimiter=",")
producto = []
for row in reader:
    producto.append(row)
ifile.close()

b = table.batch()
```

```
for j in range(1, len(productosOrden)):
    nro = productosOrden[j][0]
    idP = productosOrden[j][1]
    idC = productosOrden[j][2]
    nomC = productosOrden[j][3]
    fecO = productosOrden[j][4]
    est = productosOrden[j][5]
    cant = productosOrden[j][6]
    nomP = productosOrden[j][7]
    prec = productosOrden[j][8]
    rowkey = nro+idP+idC
    b.put(rowkey, {b'Ord:nro':nro, b'Ord:idP':idP, b'Ord:idC': idC, b'Ord:nomC':nomC, b'Ord:fecO':fecO, b'Ord:est':est, b'Ord:cant':cant, b'Ord:nomP':nomP, b'Ord:prec':prec})

    for k in range(1, len(factura)):
        nroF = factura[k][0]
        fecF = factura[k][1]
        tnro = factura[k][2]
        tpo = factura[k][3]
        nroA = factura[k][4]
        idE = factura[k][5]
        b.put(rowkey, {b'Def:nroF':nroF, b'Def:fecF':fecF, b'Def:tnro':tnro, b'Def:tpo':tpo, b'Def:nroA':nroA, b'Def:idE':idE})

#
b.put(rowkey, {b'Ord:nro':nro, b'Ord:idP':idP, b'Ord:idC': idC, b'Ord:nomC':nomC, b'Ord:fecO':fecO, b'Ord:est':est, b'Ord:cant':cant, b'Ord:nomP':nomP, b'Ord:prec':prec, b'Def:nroF':nroF, b'Def:fecF':fecF, b'Def:tnro':tnro, b'Def:tpo':tpo, b'Def:nroA':nroA, b'Def:idE':idE})
b.send()
```

1. Listar los productos del carrito de compra con estado “en proceso” de un cliente particular
Funcion python f1.py

```
bd07@serverhadoop:~$ more f1.py
import happybase
import sys

def consulta1(user):
    connection = happybase.Connection()
    table = connection.table('ARCard')
    acum = 0
    for key, data in table.scan(filter="SingleColumnValueFilter('Ord','idC',=,'binary:251')",limit=1000):
        row =table.row(key)
        if (row[b'Ord:est'])== 'EnProceso':
            acum=acum+1
            print ("IdC",data[b'Ord:idC'], "nom",data[b'Ord:nomC'], "nroO",data[b'Ord:nro'], "idP",data[b'Ord:idP'], "est",data[b'Ord:est'], "
reC",data[b'Ord:preC'], "nroF",data[b'Def:nroF'])
    print "-----TOTAL DE FILAS CONSULTADAS-----", acum
bd07@serverhadoop:~$
```

Método c1.py

```
bd07@serverhadoop:~$ more c1.py
import f1
import sys
user=sys.argv[1]
if len(sys.argv)>=2:
    f1.consulta1(user)
else:
    print "se requiere el id del cliente"
bd07@serverhadoop:~$
```

Ejecución Python c1.py 251

```
bd07@serverhadoop:~$ python c1.py 251
('IdC', '251', 'nom', 'Walter', 'nroO', '6520', 'idP', '127', 'est', 'EnProceso', 'preC', '573168', 'nroF', '4070')
('IdC', '251', 'nom', 'Walter', 'nroO', '6520', 'idP', '163', 'est', 'EnProceso', 'preC', '303126', 'nroF', '4070')
('IdC', '251', 'nom', 'Walter', 'nroO', '6520', 'idP', '166', 'est', 'EnProceso', 'preC', '326149', 'nroF', '4070')
('IdC', '251', 'nom', 'Walter', 'nroO', '6520', 'idP', '170', 'est', 'EnProceso', 'preC', '924772', 'nroF', '4070')
-----TOTAL DE FILAS CONSULTADAS----- 4
bd07@serverhadoop:~$
```

2. Calcular el valor total del carrito de compra con estado “en proceso” de un cliente particular

Funcion python f2.py

```
bd07@serverhadoop:~$ more c2.py
import f2
import sys
user=sys.argv[1]
if len(sys.argv)>=2:
    f2.consultal(user)
else:
    print "se requiere el id del cliente"
bd07@serverhadoop:~$ more f2.py
import happybase
import sys

def consultal(user):
    connection = happybase.Connection()
    table = connection.table('ARcard')
    acum = 0
    for key, data in table.scan(filter="SingleColumnValueFilter('Ord','idC',=,'binary:251')",limit=1000):
        row =table.row(key)
        if (row[b'Ord:est'])== 'EnProceso':
            #print key,row
            suma =int(data[b'Ord:preC'])
            acum = acum + suma
            print ("IdC",data[b'Ord:idC'], "nom",data[b'Ord:nomC'], "est",data[b'Ord:est'], "prec",data[b'Ord:preC'])
    print ("-----SUMA TOTAL de ORDENES ENPROCESO-----",acum)
bd07@serverhadoop:~$
```

Ejecución Python c2.py 251

```
bd07@serverhadoop:~$ python c2.py 251
('IdC', '251', 'nom', 'Walter', 'est', 'EnProceso', 'preC', '573168')
('IdC', '251', 'nom', 'Walter', 'est', 'EnProceso', 'preC', '303126')
('IdC', '251', 'nom', 'Walter', 'est', 'EnProceso', 'preC', '326149')
('IdC', '251', 'nom', 'Walter', 'est', 'EnProceso', 'preC', '924772')
('-----SUMA TOTAL de ORDENES ENPROCESO-----', 2127215)
bd07@serverhadoop:~$
```

3. Listar el código y nombre de productos incluidos en las órdenes “para envío”, junto con la cantidad a enviar

Funcion python f3.py

```
bd07@serverhadoop:~$ more f3.py
import happybase
import sys

def consultal(user):
    connection = happybase.Connection()
    table = connection.table('ARcard')
    acum = 0
    for key, data in table.scan(limit=10000):
        row =table.row(key)
        if (row[b'Ord:est'])== 'ParaEnvio':
            acum=acum+1
            print ("nro",data[b'Ord:nro'], "idP",data[b'Ord:idP'], "nomP",data[b'Ord:nomP'], "est",data[b'Ord:est'], "cat",data[b'Ord:cant'])
    print ("-----TOTAL DE FILAS CONSULTADAS-----", acum)
bd07@serverhadoop:~$
```

Ejecución Python c3.py 251

```
('nro', '6262', 'idP', '104', 'nomP', 'Limpiador 1lt', 'est', 'ParaEnvio', 'cat', '2')
('nro', '6262', 'idP', '119', 'nomP', 'Linterna JS900', 'est', 'ParaEnvio', 'cat', '7')
('nro', '6262', 'idP', '141', 'nomP', 'Tijera Raymond', 'est', 'ParaEnvio', 'cat', '4')
('nro', '6262', 'idP', '157', 'nomP', 'Aspiradora XS', 'est', 'ParaEnvio', 'cat', '2')
('nro', '6501', 'idP', '102', 'nomP', 'Pantalon Kamal', 'est', 'ParaEnvio', 'cat', '7')
('nro', '6501', 'idP', '119', 'nomP', 'Linterna JS900', 'est', 'ParaEnvio', 'cat', '4')
('nro', '6501', 'idP', '169', 'nomP', 'Licuadora EP2', 'est', 'ParaEnvio', 'cat', '1')
('nro', '6501', 'idP', '89', 'nomP', 'Licuadora GP2', 'est', 'ParaEnvio', 'cat', '1')
('nro', '6585', 'idP', '103', 'nomP', 'Aspiradora 708', 'est', 'ParaEnvio', 'cat', '2')
('nro', '6585', 'idP', '156', 'nomP', 'Carretilla Ald', 'est', 'ParaEnvio', 'cat', '8')
('nro', '6585', 'idP', '171', 'nomP', 'Televisor 60 p', 'est', 'ParaEnvio', 'cat', '2')
('nro', '6585', 'idP', '99', 'nomP', 'Kit Nash', 'est', 'ParaEnvio', 'cat', '4')
('nro', '6594', 'idP', '102', 'nomP', 'Pantalon Kamal', 'est', 'ParaEnvio', 'cat', '4')
('nro', '6594', 'idP', '103', 'nomP', 'Aspiradora 708', 'est', 'ParaEnvio', 'cat', '7')
('nro', '6594', 'idP', '147', 'nomP', 'Balon futbol', 'est', 'ParaEnvio', 'cat', '1')
('nro', '6594', 'idP', '161', 'nomP', 'Correa Louis', 'est', 'ParaEnvio', 'cat', '1')
-----TOTAL DE FILAS CONSULTADAS----- 50
bd07@serverhadoop:~$
```

4. Lista de las órdenes de un usuario particular canceladas. Listar fecha y productos de las órdenes

Funcion python f4.py

```
bd07@serverhadoop:~$ more f4.py
import happybase
import sys

def consultai(user):
    connection = happybase.Connection()
    table = connection.table('ARcard')
    acum = 0
    for key, data in table.scan(filter="SingleColumnValueFilter('Ord','idC','=', 'binary:251')", limit=1000):
        row = table.row(key)
        if (row[b'Ord:est']) == 'Cancelada':
            acum = acum + 1
            print ("idC", data[b'Ord:idC'], "nom", data[b'Ord:nomC'], "nroO", data[b'Ord:nro'], "nomP", data[b'Ord:nomP'], "est", data[b'Ord:est'],
"fecO", data[b'Ord:fecO'])
    print "-----TOTAL DE FILAS CONSULTADAS-----", acum
bd07@serverhadoop:~$
```

Ejecución Python c4.py 251

```
bd07@serverhadoop:~$ python c4.py
Traceback (most recent call last):
  File "c4.py", line 3, in <module>
    user=sys.argv[1]
IndexError: list index out of range
bd07@serverhadoop:~$ python c4.py 251
('idC', '251', 'nom', 'Walter', 'nroO', '5118', 'nomP', 'Pantalon Ahmed', 'est', 'Cancelada', 'fecO', '27/03/17')
('idC', '251', 'nom', 'Walter', 'nroO', '5118', 'nomP', 'iMac Pro', 'est', 'Cancelada', 'fecO', '27/03/17')
('idC', '251', 'nom', 'Walter', 'nroO', '5118', 'nomP', 'Estufa DF', 'est', 'Cancelada', 'fecO', '27/03/17')
('idC', '251', 'nom', 'Walter', 'nroO', '5118', 'nomP', 'ThinkPad', 'est', 'Cancelada', 'fecO', '27/03/17')
('idC', '251', 'nom', 'Walter', 'nroO', '6500', 'nomP', 'Aspiradora XS', 'est', 'Cancelada', 'fecO', '26/03/16')
('idC', '251', 'nom', 'Walter', 'nroO', '6500', 'nomP', 'Televisor 60 p', 'est', 'Cancelada', 'fecO', '26/03/16')
('idC', '251', 'nom', 'Walter', 'nroO', '6510', 'nomP', 'Vestido Marsh', 'est', 'Cancelada', 'fecO', '7/7/2017')
('idC', '251', 'nom', 'Walter', 'nroO', '6510', 'nomP', 'Carretilla Ald', 'est', 'Cancelada', 'fecO', '7/7/2017')
('idC', '251', 'nom', 'Walter', 'nroO', '6510', 'nomP', 'Equipo sonido', 'est', 'Cancelada', 'fecO', '7/7/2017')
('idC', '251', 'nom', 'Walter', 'nroO', '6510', 'nomP', 'Cama Chancellor', 'est', 'Cancelada', 'fecO', '7/7/2017')
-----TOTAL DE FILAS CONSULTADAS----- 10
bd07@serverhadoop:~$
```

5. Lista de los 5 productos más vendidos en el año 2017. Listar el nombre del producto y la cantidad vendida en 2017

Funcion python f5.py

```
bd07@serverhadoop:~$ more f5.py
import happybase
import sys

def consulta1(user):
    connection = happybase.Connection()
    table = connection.table('ARcard')
    acum = 0
    list = []
    r = 700
    for key, data in table.scan(limit=10000):
        row = table.row(key)
        cadena = (row[b'Ord:fecO'])
        if cadena.find('/17') <> -1 or cadena.find('/2017') <> -1:
            acum = acum + 1

            idP = (row[b'Ord:idP'])
            for i in (row[b'Ord:idP']):
                list = (row[b'Ord:idP'])
                sorted(list)
                #print (list)
                list.count(list)
                print ("nomP", data[b'Ord:nomP'], "fecO", data[b'Ord:fecO'])

    print "-----TOTAL DE FILAS CONSULTADAS-----", acum
bd07@serverhadoop:~$
```

Ejecución Python c5.py 251

```
('nomP', 'Cama Ralph', 'fecO', '19/01/17')
('nomP', 'Pantalon Denton', 'fecO', '19/01/17')
('nomP', 'Buso Prescott', 'fecO', '19/01/17')
('nomP', 'Camisa Gary', 'fecO', '29/12/17')
('nomP', 'Llave Allen', 'fecO', '29/12/17')
('nomP', 'Buso Prescott', 'fecO', '29/12/17')
('nomP', 'Taladro Chaim', 'fecO', '29/12/17')
('nomP', 'Linterna JS900', 'fecO', '19/06/17')
('nomP', 'Regadera Jason', 'fecO', '19/06/17')
('nomP', 'Pantalon Ahmed', 'fecO', '19/06/17')
('nomP', 'Rastrillo Kyle', 'fecO', '19/06/17')
('nomP', '280 G2 Wpro', 'fecO', '15/08/17')
('nomP', 'Pa_os cocina', 'fecO', '15/08/17')
('nomP', 'Macbook air', 'fecO', '15/08/17')
('nomP', 'Pelotas tenis', 'fecO', '15/08/17')
('nomP', 'Lenovo notebook', 'fecO', '25/02/17')
('nomP', 'Regadera Jason', 'fecO', '25/02/17')
('nomP', 'Buso Prescott', 'fecO', '25/02/17')
('nomP', 'Camiseta Camden', 'fecO', '25/02/17')
-----TOTAL DE FILAS CONSULTADAS----- 700
bd07@serverhadoop:~$
```