

```

92
93
94 SELECT c.CropName, SUM(h.QuantityKG) FROM Crop c
95 INNER JOIN Harvest h
96 ON c.CropID = h.CropID
97 GROUP BY 1
98
99
100 SELECT * FROM Fertilizer
101
102 SET max_parallel_workers_per_gather = 0;
103
104 -- Enable more parallel workers to improve parallel speed
105 SET max_parallel_workers_per_gather = 2;
106 SET max_parallel_workers = 8;

```

Data Output Messages Notifications

SQL

	cropname character varying (50)	sum numeric
1	Beans	3500.00
2	Cassava	4200.00
3	Maize	5000.00
4	Rice	0.00
5	Soybean	3000.00

```

99
100 SELECT * FROM Fertilizer
101

```

Data Output Messages Notifications

SQL

	fertilizerid [PK] integer	cropid integer	name character varying (50)	quantityused numeric (8,2)	cost numeric (10,2)	dateapplied date
1	1	1	NPK	50.00	25000.00	2025-02-01
2	2	2	Urea	40.00	20000.00	2025-03-15
3	3	3	DAP	60.00	30000.00	2025-04-10
4	4	4	Compost	70.00	15000.00	2025-03-05
5	5	5	Organic Mix	30.00	12000.00	2025-04-01
6	6	1	NPK	50.00	25000.00	2025-02-01
7	7	2	Urea	40.00	20000.00	2025-03-15
8	8	3	DAP	60.00	30000.00	2025-04-10
9	9	4	Compost	70.00	15000.00	2025-03-05
10	10	5	Organic Mix	30.00	12000.00	2025-04-01

```
109 Show config_file;  
110
```

Data Output Messages Notifications

Icons: expand, save, dropdown, clipboard, dropdown, trash, database, download, line graph, SQL

	config_file text	
1	C:/Program Files/PostgreSQL/18/data/postgresql.c...	

```
110  
111 SHOW max_parallel_workers_per_gather;  
112 SHOW max_worker_processes;  
113
```

Data Output Messages Notifications

Icons: expand, save, dropdown, clipboard, dropdown, trash, database, download, line graph, SQL

	max_worker_processes text	
1	8	

```

87      import import_foreign_schemas from branch_b
88
89  IMPORT FOREIGN SCHEMA public
90  LIMIT TO (Worker, Harvest, Sale)
91  FROM SERVER BRANCH_B_server INTO public;
92
93
94  SELECT c.CropName, SUM(h.QuantityKG) FROM Crop c
95  INNER JOIN Harvest h
96  ON c.CropID = h.CropID
97  GROUP BY 1
98
99
100 SELECT * FROM Fertilizer
101
102 SET max_parallel_workers_per_gather = 0;
103
104 -- Enable more parallel workers to improve parallel speed
105 SET max_parallel_workers_per_gather = 2;
106 SET max_parallel_workers = 8;
107 SET max_worker_processes = 8;
108
109 Show config_file;
110
111 SHOW max_parallel_workers_per_gather;
112 SHOW max_worker_processes;
113
114 SELECT * FROM Harvest
115
116 SHOW max_parallel_workers_per_gather;
117
118 EXPLAIN ANALYZE
119 SELECT c.cropname, SUM(h.quantitykg) AS total_harvest
120 FROM crop c
121 JOIN harvest h ON c.cropid = h.cropid
122 GROUP BY c.cropname:

```

```

113
114 SELECT * FROM Harvest
115
116 SHOW max_parallel_workers_per_gather;

```

Data Output Messages Notifications

SQL

Show

	harvestid integer	cropid integer	workerid integer	quantitykg numeric (10,2)	datecollected date	grade character varying (10)	buyer character varying (50)
1	1	1	1	5000.00	2025-05-21	A	AgroCo Ltd
2	2	2	2	3500.00	2025-06-16	B	FarmLink
3	3	3	3	0.00	2025-07-20	A	[null]
4	4	4	4	4200.00	2025-09-30	A	GreenMart
5	5	5	5	3000.00	2025-08-26	B	BioFarm

```

118  EXPLAIN ANALYZE
119  SELECT c.cropname, SUM(h.quantitykg) AS total_harvest
120  FROM crop c
121  JOIN harvest h ON c.cropid = h.cropid
122  GROUP BY c.cropname;

```

Data Output Messages Notifications



	QUERY PLAN	
	text	
1	GroupAggregate (cost=527.17..527.66 rows=6 width=150) (actual time=1.189..1.196 rows=5.00 loops=1)	
2	Group Key: c.cropname	
3	Buffers: shared hit=1	
4	-> Sort (cost=527.17..527.31 rows=56 width=134) (actual time=1.179..1.181 rows=5.00 loops=1)	
5	Sort Key: c.cropname	
6	Sort Method: quicksort Memory: 25kB	
7	Buffers: shared hit=1	
8	-> Hash Join (cost=101.14..525.54 rows=56 width=134) (actual time=1.150..1.155 rows=5.00 loops=1)	
9	Hash Cond: (h.cropid = c.cropid)	
10	Buffers: shared hit=1	
11	-> Foreign Scan on harvest h (cost=100.00..519.42 rows=1861 width=20) (actual time=1.074..1.076 rows=5.00 loops=1)	
12	-> Hash (cost=1.06..1.06 rows=6 width=122) (actual time=0.053..0.053 rows=11.00 loops=1)	
13	Buckets: 1024 Batches: 1 Memory Usage: 9kB	
14	Buffers: shared hit=1	
15	-> Seq Scan on crop c (cost=0.00..1.06 rows=6 width=122) (actual time=0.040..0.043 rows=11.00 loops=1)	
16	Buffers: shared hit=1	
17	Planning Time: 0.255 ms	
18	Execution Time: 2.156 ms	

```

124 --- Force parallel execution
125 EXPLAIN ANALYZE
126 SELECT /*+ Parallel */ c.cropname, SUM(h.quantitykg) AS total_harvest
127 FROM crop c
128 JOIN harvest h ON c.cropid = h.cropid
129 GROUP BY c.cropname;
130

```

Data Output Messages Notifications

SQL

QUERY PLAN	
	text
1	GroupAggregate (cost=527.17..527.66 rows=6 width=150) (actual time=1.113..1.119 rows=5.00 loops=1)
2	Group Key: c.cropname
3	Buffers: shared hit=1
4	-> Sort (cost=527.17..527.31 rows=56 width=134) (actual time=1.102..1.104 rows=5.00 loops=1)
5	Sort Key: c.cropname
6	Sort Method: quicksort Memory: 25kB
7	Buffers: shared hit=1
8	-> Hash Join (cost=101.14..525.54 rows=56 width=134) (actual time=1.077..1.081 rows=5.00 loops=1)
9	Hash Cond: (h.cropid = c.cropid)
10	Buffers: shared hit=1
11	-> Foreign Scan on harvest h (cost=100.00..519.42 rows=1861 width=20) (actual time=1.020..1.021 rows=5.00 loops=1)
12	-> Hash (cost=1.06..1.06 rows=6 width=122) (actual time=0.046..0.047 rows=11.00 loops=1)
13	Buckets: 1024 Batches: 1 Memory Usage: 9kB
14	Buffers: shared hit=1
15	-> Seq Scan on crop c (cost=0.00..1.06 rows=6 width=122) (actual time=0.038..0.040 rows=11.00 loops=1)
16	Buffers: shared hit=1
17	Planning Time: 0.192 ms
18	Execution Time: 2.149 ms

TASK 4

```

134 --- TASK 4
135
136 --- Begin the distributed transaction
137
138 BEGIN;
139 ---Operation in BRANCH A
140
141 INSERT INTO crop (fieldid, cropname, plantingdate, status, harvestdate)
142 VALUES (1, 'Tomato', '2025-05-01', 'Planted', '2025-10-27');
143
144 ---Operation in BRANCH B
145
146 INSERT INTO worker (fullname, role, contact, dailywage)
147 VALUES ('Peter Mugabo', 'Harvester', '0788000006', 3000);
148
149 ---Prepare the Transaction for 2PC
150 PREPARE TRANSACTION 'tx_insert_farm';
151
152 ---Verify Pending Distributed Transaction
153 SELECT * FROM pg_prepared_xacts;
154
155
156
157 BEGIN;
158 INSERT INTO crop (fieldid, cropname, plantingdate, status, harvestdate)
159 VALUES (1, 'Tomato T1', '2025-05-01', 'Planted', '2025-10-27');
160 INSERT INTO worker (fullname, role, contact, dailywage)
161 VALUES ('1', 'Harvester', '0788000101', 3000);
162 PREPARE TRANSACTION 'tx_farm_1';
163
164 BEGIN;
165 INSERT INTO crop (fieldid, cropname, plantingdate, status, harvestdate)
166 VALUES (2, 'Beans T2', '2025-05-02', 'Planted', '2025-10-28');

```

Data Output Messages Notifications

≡+	📄	▼	📋	▼	🗑️	🗄️	⬇️	📈	SQL
transaction xid	🔒	gid text	🔒	prepared timestamp with time zone	🔒	owner name	🔒	database name	🔒

TASK 6

```

265      THIS WILL HANG UNTIL SESSION 1 COMPLETES OR ROLLS BACK
266
267 SELECT pid, locktype, relation::regclass, mode, granted
268 FROM pg_locks
269 WHERE relation = (SELECT oid FROM pg_class WHERE relname = 'crop')
270 AND pid IN (SELECT pid FROM pg_stat_activity WHERE username = CURRENT_USER);
271
272 TASK 7

```

Data Output Messages Notifications

SQL

	pid integer	locktype text	relation regclass	mode text	granted boolean
1	25316	relation	crop	RowExclusiveLo...	true

TASK 7

```

286 -- Parallel aggregation
287 EXPLAIN ANALYZE
288 SELECT cropname, COUNT(*)
289 FROM crop
290 WHERE plantingdate >= '2025-05-01'
291 GROUP BY cropname;

```

Data Output Messages Notifications

SQL

	QUERY PLAN text
1	HashAggregate (cost=91.14..92.42 rows=128 width=126) (actual time=0.537..0.538 rows=1.00 loops=1)
2	Group Key: cropname
3	Batches: 1 Memory Usage: 32kB
4	Buffers: shared hit=84
5	-> Seq Scan on crop (cost=0.00..90.30 rows=168 width=118) (actual time=0.072..0.522 rows=1.00 loop...
6	Filter: (plantingdate >= '2025-05-01'::date)
7	Rows Removed by Filter: 10
8	Buffers: shared hit=84
9	Planning:
10	Buffers: shared hit=5
11	Planning Time: 0.157 ms
12	Execution Time: 0.585 ms

ages Notifications

fieldid integer	cropname character varying (50)	plantingdate date	harvestdate date	status character varying (20)
1	Maize	2025-01-15	2025-05-20	Harvested
2	Beans	2025-02-10	2025-06-15	Harvested
3	Rice	2025-03-01	2025-07-20	Growing
1	Cassava	2025-01-01	2025-09-30	Planted
2	Soybean	2025-03-10	2025-08-25	Planted
4	Cassava T4	2025-05-04	2025-11-10	Planted
2	Onion T8	2025-05-08	2025-12-01	Planted
1	Maize	2025-01-15	2025-05-20	Harvested
2	Beans	2025-02-10	2025-06-15	Harvested
3	Rice	2025-03-01	2025-07-20	Growing
1	Cassava	2025-01-01	2025-09-30	Planted
2	Soybean	2025-03-10	2025-08-25	Planted

TASK 8

```

304 --- TASK 8
305 -- Enable dblink extension
306 CREATE EXTENSION IF NOT EXISTS dblink;
307
308 -- Connect to a remote database (adjust connection string)
309 SELECT dblink_connect('myconn', 'dbname=BRANCH_B host=localhost port=5432 user=postgres password=Bobo1999@');
310

```

Data Output Messages Notifications

Showing rows: 1 to 1

dblink_connect text
1 OK