

AZ QUERY PLAN	
1	Hash Left Join (cost=276.26..3727.45 rows=80041 width=299) (actual time=1.987..490.059 rows=79822 loops=1)
2	Hash Cond: (cl.camplocationid = ph.camplocationcamplocationid)
3	-> Hash Join (cost=12.25..3163.27 rows=80041 width=293) (actual time=0.078..475.467 rows=79822 loops=1)
4	Hash Cond: (cl.camptypetypeid = ct.typeid)
5	-> Seq Scan on camplocation cl (cost=0.00..2932.01 rows=80041 width=79) (actual time=0.032..455.916 rows=79822 loops=1)
6	Filter: ((status)::text = 'active')::text)
7	Rows Removed by Filter: 20179
8	-> Hash (cost=11.00..11.00 rows=100 width=222) (actual time=0.031..0.033 rows=3 loops=1)
9	Buckets: 1024 Batches: 1 Memory Usage: 9kB
10	-> Seq Scan on camptype ct (cost=0.00..11.00 rows=100 width=222) (actual time=0.024..0.025 rows=3 loops=1)
11	-> Hash (cost=264.00..264.00 rows=1 width=10) (actual time=1.869..1.870 rows=0 loops=1)
12	Buckets: 1024 Batches: 1 Memory Usage: 8kB
13	-> Seq Scan on pricehistory ph (cost=0.00..264.00 rows=1 width=10) (actual time=1.869..1.869 rows=0 loops=1)
14	Filter: ((CURRENT_DATE >= date_from) AND (CURRENT_DATE <= date_to))
15	Rows Removed by Filter: 10000
16	Planning Time: 0.795 ms
17	Execution Time: 493.095 ms

- **View2: view_all_reservations**

1. Примарен филтер ќе биде според **reservationId**, а дополнително може да се пребарува според: guest_id, host_id, campLocationId, reservation_status, check_in_date и check_out_date
2. Примарен случај на употреба е преглед на сите резервации од страна на корисник (guest) и преглед на резервации за одреден хост.
3. Иницијално време на извршување е **83ms**, ова време на извршување е прифатливо, затоа нема потреба од индексирање.

explain analyze select * from view_all_reservations

Step	Operation	Cost	Rows	Width	Actual Time	Actual Rows	Loops
1	Nested Loop Left Join	1.14	17.28	132	83.340	83.351	1
2	-> Nested Loop	0.71	16.75	96	19.449	19.453	1
3	-> Index Scan using reservation_pkey on reservation r	0.42	8.44	39	18.850	18.853	1
4	Index Cond: (reservationid = 100)						
5	-> Index Scan using camplocation_pkey on camplocation cl	0.29	8.31	61	0.588	0.588	1
6	Index Cond: (camplocationid = r.camplocationid)						
7	-> Index Only Scan using host_host_pkey on host_host hh	0.42	0.48	8	63.830	63.833	1
8	Index Cond: (camplocationid = cl.camplocationid)						
9	Heap Fetches: 0						
10	Planning Time: 1026.351 ms						
11	Execution Time: 83.417 ms						

- **View3: view_available_activities**

1. Примарен филтер ќе биде според **campLocationId**, што му овозможува на корисникот да ги филтрира активностите специфични за локацијата која ја разгледува, дополнително може да се пребарува и по име на активността.
2. Примарен случај на употреба е преглед на сите достапни и активни активности за одреден камп.
3. Иницијално време на извршување е **9s**, ова време не е прифатливо, затоа пристапуваме кон правење на индекси.

explain analyze select * from view_available_activities

Step	Operation	Cost	Rows	Width	Actual Time	Actual Rows	Loops
1	Nested Loop	1000.71	4565.71	84	964.135	9716.040	1
2	-> Index Scan using camplocation_pkey on camplocation cl	0.29	8.31	23	70.879	70.883	1
3	Index Cond: (camplocationid = 2000)						
4	Filter: ((status)::text = 'active':text)						
5	-> Gather	1000.42	4557.36	65	893.250	9645.145	1
6	Workers Planned: 1						
7	Workers Launched: 1						
8	-> Nested Loop	0.42	3556.96	65	5212.259	9588.141	1
9	-> Parallel Seq Scan on activity_activity aa	0.00	3540.08	8	5059.317	9351.483	1
10	Filter: (camplocationid = 2000)						
11	Rows Removed by Filter: 150284						
12	-> Index Scan using activity_pkey on activity a	0.42	8.44	61	236.641	236.642	1
13	Index Cond: (activity_id = aa.activityid2)						
14	Planning Time: 1378.304 ms						
15	Execution Time: 9716.109 ms						

4. Најбавната операција е *Parallel Seq Scan* за *activity_activity*

explain analyze select * from view_available_a

Grid	Text
1	Nested Loop (cost=1000.71..4565.71 rows=4 width=84) (actual time=964.135..9716.040 rows=2 loops=1)
2	-> Index Scan using camplocation_pkey on camplocation cl (cost=0.29..8.31 rows=1 width=23) (actual time=70.879..70.883 row
3	Index Cond: (camplocationid = 2000)
4	Filter: ((status)::text = 'active'::text)
5	-> Gather (cost=1000.42..4557.36 rows=4 width=65) (actual time=893.250..9645.145 rows=2 loops=1)
6	Workers Planned: 1
7	Workers Launched: 1
8	-> Nested Loop (cost=0.42..3556.96 rows=2 width=65) (actual time=5212.259..9588.141 rows=1 loops=2)
9	-> Parallel Seq Scan on activity_activity aa (cost=0.00..3540.08 rows=2 width=8) (actual time=5059.317..9351.483 rows=1
10	Filter: (camplocationcamplocationid = 2000)
11	Rows Removed by Filter: 150284
12	-> Index Scan using activity_pkey on activity a (cost=0.42..8.44 rows=1 width=61) (actual time=236.641..236.642 rows=1
13	Index Cond: (activity_id = aa.activityactivity_id2)
14	Planning Time: 1378.304 ms
15	Execution Time: 9716.109 ms

Времето изминато за извршување на операцијата **insert** изнесува:

```
INSERT INTO Activity_Activity (  
Activityactivity_id2,  
CampLocationcampLocationId )  
VALUES (1, 10);
```

QUERY PLAN

1	Insert on activity_activity (cost=0.00..0.01 rows=0 width=0) (actual time=494.857..494.858 rows=0 loops=1)
2	-> Result (cost=0.00..0.01 rows=1 width=8) (actual time=0.003..0.003 rows=1 loops=1)
3	Planning Time: 0.032 ms
4	Trigger for constraint activity_activity_activityactivity_id2_fkey: time=212.440 calls=1
5	Trigger for constraint activity_activity_camplocationcamplocationid_fkey: time=1444.693 calls=1
6	Execution Time: 2152.022 ms

Времето изминато за извршување на операцијата **update** изнесува

```
UPDATE Activity_Activity  
SET CampLocationcampLocationId = 11  
WHERE Activityactivity_id2 = 1 AND  
CampLocationcampLocationId = 10;
```

explain analyze UPDATE Activity_Activity SET C | Enter a SQL expression to filter results (use Ctrl+Space)

Grid	Text
1	Update on activity_activity (cost=0.42..8.44 rows=0 width=0) (actual time=1646.454..1646.455 rows=0 loops=1)
2	-> Index Scan using activity_activity_pkey on activity_activity (cost=0.42..8.44 rows=1 width=10) (actual time=0.145..0.148 rows=
3	Index Cond: ((activityactivity_id2 = 1) AND (camplocationcamplocationid = 10))
4	Planning Time: 0.120 ms
5	Trigger for constraint activity_activity_camplocationcamplocationid_fkey: time=0.262 calls=1
6	Execution Time: 1670.199 ms

5. Времето на извршување со индекси изнесува

```
CREATE INDEX idx_aa_location_activity
ON Activity_Activity(CampLocationcampLocationId,
Activityactivity_id2);
```

QUERY PLAN

1	Nested Loop (cost=1.14..50.60 rows=4 width=84) (actual time=0.206..0.246 rows=2 loops=1)
2	-> Index Scan using camplocation_pkey on camplocation cl (cost=0.29..8.31 rows=1 width=23) (actual time=0.071..0.072 rows=1 loops=1)
3	Index Cond: (camplocationid = 2000)
4	Filter: ((status)::text = 'active')::text)
5	-> Nested Loop (cost=0.84..42.24 rows=4 width=65) (actual time=0.132..0.171 rows=2 loops=1)
6	-> Index Only Scan using idx_aa_location_activity on activity_activity aa (cost=0.42..8.49 rows=4 width=8) (actual time=0.099..0.101 rows=2 loops=1)
7	Index Cond: (camplocationcamplocationid = 2000)
8	Heap Fetches: 0
9	-> Index Scan using activity_pkey on activity a (cost=0.42..8.44 rows=1 width=61) (actual time=0.032..0.032 rows=1 loops=2)
10	Index Cond: (activity_id = aa.activityactivity_id2)
11	Planning Time: 1.440 ms
12	Execution Time: 0.284 ms

6. Времето на извршување на операцијата *Insert по индексирање* изнесува

```
INSERT INTO Activity_Activity (
Activityactivity_id2,
CampLocationcampLocationId )
VALUES (1, 10);
```

explain analyze INSERT INTO Activity_Activity (| Enter a SQL expression to filter results (use Ctrl+Space)

Grid	Text
1	Insert on activity_activity (cost=0.00..0.01 rows=0 width=0) (actual time=0.256..0.257 rows=0 loops=1)
2	-> Result (cost=0.00..0.01 rows=1 width=8) (actual time=0.003..0.003 rows=1 loops=1)
3	Planning Time: 0.033 ms
4	Trigger for constraint activity_activity_activityactivity_id2_fkey: time=0.361 calls=1
5	Trigger for constraint activity_activity_camplocationcamplocationid_fkey: time=0.198 calls=1
6	Execution Time: 0.847 ms

Времето на извршување на операцијата **Update по индексирање** изнесува

```
UPDATE Activity_Activity
SET CampLocationcampLocationId =12
WHERE Activityactivity_id2 = 1 AND
CampLocationcampLocationId = 10;
```

Grid		asc QUERY PLAN
1		Update on activity_activity (cost=0.42..8.44 rows=0 width=0) (actual time=0.257..0.258 rows=0 loops=1)
2		-> Index Scan using idx_aa_location_activity on activity_activity (cost=0.42..8.44 rows=1 width=10) (actual time=0.167..0.168 rows=1 loops=1)
3		Index Cond: ((camplocationcamplocationid = 10) AND (activityactivity_id2 = 1))
4		Planning Time: 0.130 ms
5		Trigger for constraint activity_activity_camplocationcamplocationid_fkey: time=0.211 calls=1
6		Execution Time: 0.504 ms

- **View4: view_favorite_camps**

1. Примарниот филтер ќе биде според **GuestUseruserId**, при што со погледот се прикажуваат омилените камп локации за конкретен корисник заедно со детали за кампот.
2. Примарниот случај на употреба е прикажување на омилените камп локации за конкретен корисник преку филтрирање според GuestUseruserId.
3. Иницијално време на извршување е **12.5 s**, ова време не е прифатливо, затоа пристапуваме кон правење на индекси.

Grid		asc QUERY PLAN
1		Nested Loop (cost=1000.72..9679.38 rows=1 width=83) (actual time=12547.552..12552.341 rows=0 loops=1)
2		-> Nested Loop (cost=1000.42..9679.04 rows=1 width=16) (actual time=12547.551..12552.339 rows=0 loops=1)
3		-> Gather (cost=1000.00..9670.60 rows=1 width=8) (actual time=12547.550..12552.336 rows=0 loops=1)
4		Workers Planned: 2
5		Workers Launched: 2
6		-> Parallel Seq Scan on favorites_guest_saves fg (cost=0.00..8670.50 rows=1 width=8) (actual time=12479.495..12479.496 rows=0 loops=1)
7		Filter: (guestuserid = 50000)
8		Rows Removed by Filter: 300000
9		-> Index Scan using favorites_pkey on favorites f (cost=0.42..8.44 rows=1 width=12) (never executed)
10		Index Cond: (favorite_id = fg.favoritesfavoriteid)
11		-> Index Scan using camplocation_pkey on camplocation cl (cost=0.29..0.34 rows=1 width=71) (never executed)
12		Index Cond: (camplocationid = f.locationid)
13		Planning Time: 2205.313 ms
14		Execution Time: 12552.413 ms

4. Најбавната операција е *Parallel Seq Scan* за *favorites_guest_saves*

explain analyze select * from view_favorite_car | Enter a SQL expression to filter results (use Ctrl+Space)

Grid	ABC QUERY PLAN
1	Nested Loop (cost=1000.72..9679.38 rows=1 width=83) (actual time=12547.552..12552.341 rows=0 loops=1)
2	-> Nested Loop (cost=1000.42..9679.04 rows=1 width=16) (actual time=12547.551..12552.339 rows=0 loops=1)
3	-> Gather (cost=1000.00..9670.60 rows=1 width=8) (actual time=12547.550..12552.336 rows=0 loops=1)
4	Workers Planned: 2
5	Workers Launched: 2
6	-> Parallel Seq Scan on favorites_guest_saves fg (cost=0.00..8670.50 rows=1 width=8) (actual time=12479.495..12479.496 rows=0 loops=1)
7	Filter: (guestuserid = 50000)
8	Rows Removed by Filter: 300000
9	-> Index Scan using favorites_pkey on favorites f (cost=0.42..8.44 rows=1 width=12) (never executed)
10	Index Cond: (favorite_id = fg.favoritesfavoriteid)
11	-> Index Scan using camplocation_pkey on camplocation cl (cost=0.29..0.34 rows=1 width=71) (never executed)
12	Index Cond: (camplocationid = f.locationid)
13	Planning Time: 2205.313 ms
14	Execution Time: 12552.413 ms

Времето изминато за извршување на операцијата **insert** изнесува :

```
explain analyze
INSERT INTO
Favorites_Guest_saves (
Favoritesfavoriteid,
GuestUseruserid)
VALUES (1, 55492);
```

ABC QUERY PLAN
Insert on favorites_guest_saves (cost=0.00..0.01 rows=0 width=0) (actual time=0.763..0.764 rows=0 loops=1)
-> Result (cost=0.00..0.01 rows=1 width=8) (actual time=0.001..0.002 rows=1 loops=1)
Planning Time: 0.039 ms
Trigger for constraint favorites_guest_saves_favoritesfavoriteid_fkey: time=18.058 calls=1
Trigger for constraint favorites_guest_saves_guestuserid_fkey: time=589.906 calls=1
Execution Time: 608.772 ms

Времето изминато за извршување на операцијата **update** изнесува

```
explain analyze
UPDATE Favorites_Guest_saves
SET GuestUseruserid = 6
WHERE Favoritesfavoriteid = 1
AND GuestUseruserid = 55492;
```


ABC QUERY PLAN	
1	Update on favorites_guest_saves (cost=0.42..8.45 rows=0 width=0) (actual time=197.040..197.041 rows=0 loops=1)
2	-> Index Scan using favorites_guest_saves_pkey on favorites_guest_saves (cost=0.42..8.45 rows=1 width=10) (actual time=0.234..0.237 rows=1 loops=1)
	Index Cond: ((favoritesfavoriteid = 1) AND (guestuserid = 55492))
	Planning Time: 0.119 ms
	Trigger for constraint favorites_guest_saves_guestuserid_fkey: time=0.379 calls=1
	Execution Time: 197.458 ms

5. Времето на извршување со индекси изнесува

```
CREATE INDEX idx_fgs_composite
ON Favorites_Guest_saves (GuestUserId, Favoritesfavoriteid);
```

ABC QUERY PLAN	
1	Nested Loop (cost=1.14..17.22 rows=1 width=83) (actual time=22.757..22.760 rows=0 loops=1)
2	-> Nested Loop (cost=0.85..16.88 rows=1 width=16) (actual time=22.757..22.759 rows=0 loops=1)
3	-> Index Only Scan using idx_fgs_composite on favorites_guest_saves fg (cost=0.42..8.44 rows=1 width=8) (actual time=22.756..22.757 rows=0 loops=1)
4	Index Cond: (guestuserid = 50000)
5	Heap Fetches: 0
6	-> Index Scan using favorites_pkey on favorites f (cost=0.42..8.44 rows=1 width=12) (never executed)
7	Index Cond: (favorite_id = fg.favoritesfavoriteid)
8	-> Index Scan using camplocation_pkey on camplocation cl (cost=0.29..0.34 rows=1 width=71) (never executed)
9	Index Cond: (camplocationid = f.locationid)
10	Planning Time: 3.436 ms
11	Execution Time: 22.802 ms

6. Времето на извршување на операцијата *Insert по индексирање* изнесува

```
explain analyze
INSERT INTO
Favorites_Guest_saves (
Favoritesfavoriteid,
GuestUserId)
VALUES (1, 55492);
```

ABC QUERY PLAN	
1	Insert on favorites_guest_saves (cost=0.00..0.01 rows=0 width=0) (actual time=296.379..296.379 rows=0 loops=1)
2	-> Result (cost=0.00..0.01 rows=1 width=8) (actual time=0.002..0.003 rows=1 loops=1)
3	Planning Time: 0.033 ms
4	Trigger for constraint favorites_guest_saves_favoritesfavoriteid_fkey: time=0.236 calls=1
5	Trigger for constraint favorites_guest_saves_guestuserid_fkey: time=0.204 calls=1
6	Execution Time: 296.850 ms

Времето на извршување на операцијата **Update по индексирање** изнесува

```
explain analyze
UPDATE Favorites_Guest_saves
SET GuestUseruserId = 6
WHERE FavoritesfavoriteId = 1
AND GuestUseruserId = 55492;
```

ABC QUERY PLAN	
1	Update on favorites_guest_saves (cost=0.42..8.45 rows=0 width=0) (actual time=0.365..0.366 rows=0 loops=1)
2	-> Index Scan using idx_fgs_composite on favorites_guest_saves (cost=0.42..8.45 rows=1 width=10) (actual time
3	Index Cond: ((guestuseruserid = 55492) AND (favoritesfavoriteid = 1))
4	Planning Time: 0.137 ms
5	Trigger for constraint favorites_guest_saves_guestuseruserid_fkey: time=0.153 calls=1
6	Execution Time: 0.586 ms

- **View5: view_host_total_earnings**

1. Примарен филтер ќе биде според **host id**, а дополнително може да се анализира заработката според total_payments и total_earnings.
2. Примарен случај на употреба е аналитички преглед на вкупната заработка на секој хост, како и увид во бројот на реализирани плаќања од сите камп локации кои се поврзани со тој хост.
3. Бидејќи ова query е **аналитичко нема да се прави индексирање**. Иницијалното време на извршување изнесува 21.919ms.

QUERY PLAN	
7	-> Parallel Seq Scan on payment p (cost=0.00..11148.33 rows=83445 width=13) (actual time=0.443..0.444 rows=1 loops=3)
8	Filter: ((payment_status)::text = 'completed'::text)
9	Rows Removed by Filter: 2
10	-> Hash (cost=3394.18..3394.18 rows=2 width=8) (actual time=0.073..0.077 rows=0 loops=3)
11	Buckets: 1024 Batches: 1 Memory Usage: 8kB
12	-> Nested Loop (cost=0.86..3394.18 rows=2 width=8) (actual time=0.073..0.076 rows=0 loops=3)
13	Join Filter: (hh.camplocationcamplocationid = r.camplocationcamplocationid)
14	-> Nested Loop (cost=0.58..3393.63 rows=1 width=12) (actual time=0.072..0.074 rows=0 loops=3)
15	-> Nested Loop (cost=0.29..3389.32 rows=1 width=8) (actual time=0.071..0.072 rows=0 loops=3)
16	-> Index Only Scan using host_pkey on host h (cost=0.29..4.31 rows=1 width=4) (actual time=0.071..0.071 rows=0 loops=3)
17	Index Cond: (userid = 200)
18	Heap Fetches: 0
19	-> Seq Scan on host_host hh (cost=0.00..3385.00 rows=1 width=8) (never executed)
20	Filter: (hostuserid = 200)
21	-> Index Only Scan using camplocation_pkey on camplocation cl (cost=0.29..4.31 rows=1 width=4) (never executed)
22	Index Cond: (camplocationid = hh.camplocationcamplocationid)
23	Heap Fetches: 0
24	-> Index Scan using reservation_camplocationcamplocationid_daterange_excl on reservation r (cost=0.28..0.51 rows=3 width=8) (never executed)
25	Index Cond: (camplocationcamplocationid = cl.camplocationid)
26	Planning Time: 10.634 ms
27	Execution Time: 21.919 ms

- **View6: view_camp_avg_rating**

1. Примарен филтер ќе биде според **campLocationId**, што овозможува брз приказ на оцените за конкретна локација, а дополнително погледот може да се користи за рангирање на камповите според нивниот просечен успех.

2. Примарен случај на употреба е преглед на просечниот рејтинг и вкупниот број на рецензии за секој камп.

3. Иницијално време на извршување е **3709 ms**, ова време не е прифатливо, но бидејќи станува збор за **аналитичко** query не пристапуваме кон индексирање.

RDB QUERY PLAN	
1	GroupAggregate (cost=0.29..2230.35 rows=1 width=63) (actual time=3709.104..3709.107 rows=1 loops=1)
2	-> Nested Loop Left Join (cost=0.29..2230.33 rows=2 width=31) (actual time=418.973..3709.077 rows=2 loops=1)
3	-> Index Scan using camplocation_pkey on camplocation cl (cost=0.29..8.31 rows=1 width=23) (actual time=26.807..26.814 rows=1 loops=1)
4	Index Cond: (camplocationid = 13000)
5	-> Seq Scan on review r (cost=0.00..2222.00 rows=2 width=12) (actual time=392.155..3682.249 rows=2 loops=1)
6	Filter: (camplocationcamplocationid = 13000)
7	Rows Removed by Filter: 99998
8	Planning Time: 39.061 ms
9	Execution Time: 3709.293 ms

- **View7: view_available_equipment**

1. Примарен филтер ќе биде според campLocationId.
2. Примарната употреба на погледот е филтрирање според камп локацијата (преку campLocationId) а притоа се прикажуваат и дополнителни информации за тип, количина и статус на достапност.
3. Иницијално време на извршување е **39172 ms** , ова време не е прифатливо, затоа пристапуваме кон правење на индекси.

	ABC QUERY PLAN
1	Nested Loop (cost=1000.29..24530.25 rows=8 width=357) (actual time=39167.707..39172.859 rows=8 loops=1)
2	-> Index Scan using camplocation_pkey on camplocation cl (cost=0.29..8.31 rows=1 width=23) (actual time=32.580..32.586 rows=1 loops=1)
3	Index Cond: (camplocationid = 58000)
4	-> Nested Loop (cost=1000.00..24521.82 rows=8 width=306) (actual time=39135.119..39140.260 rows=8 loops=1)
5	Join Filter: (et.equipmenttypeid = e.equipmenttypeequipmenttypeid)
6	Rows Removed by Join Filter: 216
7	-> Seq Scan on equipmenttype et (cost=0.00..11.00 rows=100 width=222) (actual time=144.216..144.249 rows=28 loops=1)
8	-> Materialize (cost=1000.00..24498.84 rows=8 width=92) (actual time=1198.699..1392.713 rows=8 loops=28)
9	-> Gather (cost=1000.00..24498.80 rows=8 width=92) (actual time=33563.554..38995.935 rows=8 loops=1)
10	Workers Planned: 2
11	Workers Launched: 2
12	-> Parallel Seq Scan on equipment e (cost=0.00..23498.00 rows=3 width=92) (actual time=16106.488..38835.189 rows=3 loops=3)
13	Filter: (((camplocationcamplocationid = 58000) AND ((is_available)::text = 'yes'::text)))
14	Rows Removed by Filter: 333331
15	Planning Time: 406.553 ms
16	Execution Time: 39172.924 ms

4. Најбавната операција е **Parallel Seq Scan за equipment**

	ABC QUERY PLAN
1	Nested Loop (cost=1000.29..24530.25 rows=8 width=357) (actual time=39167.707..39172.859 rows=8 loops=1)
2	-> Index Scan using camplocation_pkey on camplocation cl (cost=0.29..8.31 rows=1 width=23) (actual time=32.580..32.586 rows=1 loops=1)
3	Index Cond: (camplocationid = 58000)
4	-> Nested Loop (cost=1000.00..24521.82 rows=8 width=306) (actual time=39135.119..39140.260 rows=8 loops=1)
5	Join Filter: (et.equipmenttypeid = e.equipmenttypeequipmenttypeid)
6	Rows Removed by Join Filter: 216
7	-> Seq Scan on equipmenttype et (cost=0.00..11.00 rows=100 width=222) (actual time=144.216..144.249 rows=28 loops=1)
8	-> Materialize (cost=1000.00..24498.84 rows=8 width=92) (actual time=1198.699..1392.713 rows=8 loops=28)
9	-> Gather (cost=1000.00..24498.80 rows=8 width=92) (actual time=33563.554..38995.935 rows=8 loops=1)
10	Workers Planned: 2
11	Workers Launched: 2
12	-> Parallel Seq Scan on equipment e (cost=0.00..23498.00 rows=3 width=92) (actual time=16106.488..38835.189 rows=3 loops=3)
13	Filter: (((camplocationcamplocationid = 58000) AND ((is_available)::text = 'yes'::text)))
14	Rows Removed by Filter: 333331
15	Planning Time: 406.553 ms
16	Execution Time: 39172.924 ms

Времето изминато за извршување на операцијата **insert** изнесува :

```
INSERT INTO Equipment
(EquipmentTypeEquipmentId,CampLocationcampLocationId,
name,
description,
total_quantity,
available_quantity,
rental_price_per_day,
deposit_amount,
condition_status,
is_available)

VALUES (1,10,'Tent
Deluxe',          '4-
person waterproof
tent',10,10,15.50,50.00,'n
ew','yes');
```

	ABC QUERY PLAN
1	Insert on equipment (cost=0.00..0.01 rows=0 width=0) (actual time=2677.416..2677.417 rows=0 loops=1)
2	-> Result (cost=0.00..0.01 rows=1 width=1158) (actual time=679.608..679.610 rows=1 loops=1)
3	Planning Time: 0.059 ms
4	Trigger for constraint equipment_equipments_typeequipment_type_id_fkey: time=0.826 calls=1
5	Trigger for constraint equipment_camplocationcamplocation_id_fkey: time=0.348 calls=1
6	Execution Time: 2678.621 ms

Времето изминато за извршување на операцијата **update** изнесува :

```
UPDATE Equipment SET name =
'Tent Premium',
rental_price_per_day = 20.00,
condition_status = 'good' WHERE
equipmentId = 1;
```

	ABC QUERY PLAN
1	Update on equipment (cost=0.42..8.44 rows=0 width=0) (actual time=176.468..176.470 rows=0 loops=1)
2	-> Index Scan using equipment_pkey on equipment (cost=0.42..8.44 rows=1 width=596) (actual time=176.466..176.467 rows=0 loops=1)
3	Index Cond: (equipmentid = 1)
4	Planning Time: 0.128 ms
5	Execution Time: 176.623 ms

5. Времето на извршување со индекси изнесува

```
CREATE INDEX
idx_equipment_camp_available ON
Equipment
(CampLocationcampLocationId,
is_available);
```

	ABC QUERY PLAN
1	Nested Loop (cost=17.05..56.81 rows=8 width=357) (actual time=0.208..0.321 rows=8 loops=1)
2	-> Index Scan using camplocation_pkey on camplocation cl (cost=0.29..8.31 rows=1 width=23) (actual time=0.065..0.067 rows=1 loops=1)
3	Index Cond: (camplocationid = 58000)
4	-> Hash Join (cost=16.76..48.38 rows=8 width=306) (actual time=0.137..0.245 rows=8 loops=1)
5	Hash Cond: (e.equipmenttypeequipmenttypeid = et.equipmenttypeid)
6	-> Bitmap Heap Scan on equipment e (cost=4.51..36.11 rows=8 width=92) (actual time=0.077..0.177 rows=8 loops=1)
7	Recheck Cond: ((camplocationcamplocationid = 58000) AND (((is_available)::text = 'yes'::text)))
8	Heap Blocks: exact=8
9	-> Bitmap Index Scan on idx_equipment_camp_available (cost=0.00..4.50 rows=8 width=0) (actual time=0.048..0.048 rows=8 loops=1)
10	Index Cond: ((camplocationcamplocationid = 58000) AND (((is_available)::text = 'yes'::text)))
11	-> Hash (cost=11.00..11.00 rows=100 width=222) (actual time=0.045..0.046 rows=28 loops=1)
12	Buckets: 1024 Batches: 1 Memory Usage: 10kB
13	-> Seq Scan on equipmenttype et (cost=0.00..11.00 rows=100 width=222) (actual time=0.031..0.035 rows=28 loops=1)
14	Planning Time: 0.692 ms
15	Execution Time: 0.378 ms

6.

Времето на извршување на операцијата *Insert по индексирање* изнесува

```
INSERT INTO Equipment (
EquipmentTypeEquipmentTypeId,
CampLocationcampLocationId,
name, description, total_quantity,
available_quantity,
rental_price_per_day,
deposit_amount,
condition_status,
is_available)
VALUES (
1, 10, 'Tent Deluxe',
'4-person waterproof tent', 10, 10, 15.50,
50.00, 'new', 'yes');
```

	ABC QUERY PLAN
1	Insert on equipment (cost=0.00..0.01 rows=0 width=0) (actual time=213.099..213.100 rows=0 loops=1)
2	-> Result (cost=0.00..0.01 rows=1 width=1158) (actual time=0.056..0.058 rows=1 loops=1)
3	Planning Time: 0.080 ms
4	Trigger for constraint equipment_equipmenttypeequipmenttypeid_fkey; time=0.269 calls=1
5	Trigger for constraint equipment_camplocationcamplocationid_fkey; time=0.178 calls=1
6	Execution Time: 213.578 ms

Времето на извршување на операцијата **update по индексирање** изнесува

```
UPDATE Equipment SET name =
'Tent Premium',
rental_price_per_day = 20.00,
condition_status = 'good' WHERE
equipmentId = 1;
```

	ABC QUERY PLAN
1	Update on equipment (cost=0.42..8.44 rows=0 width=0) (actual time=0.102..0.103 rows=0 loops=1)
2	-> Index Scan using equipment_pkey on equipment (cost=0.42..8.44 rows=1 width=596) (actual time=0.101..0.101 rows=0 loops=1)
3	Index Cond: (equipmentid = 1)
4	Planning Time: 0.137 ms
5	Execution Time: 0.163 ms

- **View8: view_reservation_payment_status**

1. Примарен филтер ќе биде според **guest_id**, што овозможува брз приказ на статусот на плаќањата за конкретен гостин, а дополнително погледот може да се користи за идентификување на резервации со нецелосно или без плаќање.
2. Примарен случај на употреба е преглед на вкупно платениот износ, преостанатиот долг и статусот на плаќање (paid, partially_paid, not_paid) за секоја резервација.
3. Иницијалното време на извршување изнесува 133ms, што значи нема потреба од индексирање.

5	Sort Method: quicksort Memory: 25kB
6	-> Gather (cost=5056.54..20432.13 rows=2 width=41) (actual time=130.955..133.716 rows=1 loops=1)
7	Workers Planned: 2
8	Workers Launched: 2
9	-> Nested Loop (cost=4056.54..19431.93 rows=1 width=41) (actual time=102.339..125.667 rows=0 loops=3)
10	-> Parallel Hash Right Join (cost=4056.25..19423.62 rows=1 width=22) (actual time=100.394..123.720 rows=0 loops=3)
11	Hash Cond: (p.reservationreservationid = r.reservationid)
12	-> Parallel Seq Scan on payment p (cost=0.00..15148.33 rows=83445 width=9) (actual time=0.072..86.351 rows=66593 loops=3)
13	Filter: ((payment_status)::text = 'completed'::text)
14	Rows Removed by Filter: 266740
15	-> Parallel Hash (cost=4056.24..4056.24 rows=1 width=17) (actual time=31.754..31.756 rows=0 loops=3)
16	Buckets: 1024 Batches: 1 Memory Usage: 40kB
17	-> Parallel Seq Scan on reservation r (cost=0.00..4056.24 rows=1 width=17) (actual time=22.570..31.535 rows=0 loops=3)
18	Filter: (guestuserid = 200)
19	Rows Removed by Filter: 71138
20	-> Index Scan using camplocation_pkey on camplocation cl (cost=0.29..8.31 rows=1 width=23) (actual time=5.816..5.817 rows=1 loops=1)
21	Index Cond: (camplocationid = r.camplocationcamplocationid)
22	Planning Time: 1.497 ms
23	Execution Time: 133.883 ms

- **View9: view_popular_camps**

1. Примарен филтер ќе биде според campLocationId, а дополнително погледот ќе овозможи пребарување и филтрирање според пресметаниот статус popularity_status за да се идентификуваат најдобро рангираните локации.
2. Примарен случај на употреба е преглед на сите камп локации заедно со нивните статистички податоци за резервации и оценки, со цел автоматска категоризација на секој камп како „top_rated“, „popular“ или „standard“.
3. Иницијално време на извршување е **7672 ms**, ова време не е прифатливо, затоа пристапуваме кон правење на индекси.

	ABC QUERY PLAN
1	GroupAggregate (cost=7741.57..7741.73 rows=1 width=141) (actual time=7667.508..7672.149 rows=1 loops=1)
2	-> Sort (cost=7741.57..7741.61 rows=14 width=73) (actual time=7667.471..7672.112 rows=8 loops=1)
3	Sort Key: r.reservationid
4	Sort Method: quicksort Memory: 25kB
5	-> Nested Loop Left Join (cost=1004.60..7741.30 rows=14 width=73) (actual time=6799.340..7672.089 rows=8 loops=1)
6	-> Nested Loop Left Join (cost=4.60..20.39 rows=2 width=69) (actual time=35.217..35.225 rows=1 loops=1)
7	-> Index Scan using camplocation_pkey on camplocation cl (cost=0.29..8.31 rows=1 width=61) (actual time=0.811..0.818 rows=1 loops=1)
8	Index Cond: (camplocationid = 100)
9	-> Bitmap Heap Scan on review rv (cost=4.31..12.06 rows=2 width=12) (actual time=34.398..34.399 rows=0 loops=1)
10	Recheck Cond: (camplocationcamplocationid = 100)
11	-> Bitmap Index Scan on idx_review_camp_date (cost=0.00..4.31 rows=2 width=0) (actual time=34.391..34.391 rows=0 loops=1)
12	Index Cond: (camplocationcamplocationid = 100)
13	-> Materialize (cost=1000.00..7720.76 rows=7 width=8) (actual time=6764.119..7636.855 rows=8 loops=1)
14	-> Gather (cost=1000.00..7720.72 rows=7 width=8) (actual time=6764.112..7636.840 rows=8 loops=1)
15	Workers Planned: 2
16	Workers Launched: 2
17	-> Parallel Seq Scan on reservation r (cost=0.00..6720.02 rows=3 width=8) (actual time=5521.248..7624.680 rows=3 loops=3)
18	Filter: (camplocationcamplocationid = 100)
19	Rows Removed by Filter: 144767
20	Planning Time: 396.548 ms
21	Execution Time: 7672.254 ms

4. Најбавната операција е *Parallel Seq Scan за reservation*

	ABC QUERY PLAN
1	GroupAggregate (cost=7741.57..7741.73 rows=1 width=141) (actual time=7667.508..7672.149 rows=1 loops=1)
2	-> Sort (cost=7741.57..7741.61 rows=14 width=73) (actual time=7667.471..7672.112 rows=8 loops=1)
3	Sort Key: r.reservationid
4	Sort Method: quicksort Memory: 25kB
5	-> Nested Loop Left Join (cost=1004.60..7741.30 rows=14 width=73) (actual time=6799.340..7672.089 rows=8 loops=1)
6	-> Nested Loop Left Join (cost=4.60..20.39 rows=2 width=69) (actual time=35.217..35.225 rows=1 loops=1)
7	-> Index Scan using camplocation_pkey on camplocation cl (cost=0.29..8.31 rows=1 width=61) (actual time=0.811..0.818 rows=1 loops=1)
8	Index Cond: (camplocationid = 100)
9	-> Bitmap Heap Scan on review rv (cost=4.31..12.06 rows=2 width=12) (actual time=34.398..34.399 rows=0 loops=1)
10	Recheck Cond: (camplocationcamplocationid = 100)
11	-> Bitmap Index Scan on idx_review_camp_date (cost=0.00..4.31 rows=2 width=0) (actual time=34.391..34.391 rows=0 loops=1)
12	Index Cond: (camplocationcamplocationid = 100)
13	-> Materialize (cost=1000.00..7720.76 rows=7 width=8) (actual time=6764.119..7636.855 rows=8 loops=1)
14	-> Gather (cost=1000.00..7720.72 rows=7 width=8) (actual time=6764.112..7636.840 rows=8 loops=1)
15	Workers Planned: 2
16	Workers Launched: 2
17	-> Parallel Seq Scan on reservation r (cost=0.00..6720.02 rows=3 width=8) (actual time=5521.248..7624.680 rows=3 loops=3)
18	Filter: (camplocationcamplocationid = 100)
19	Rows Removed by Filter: 144767
20	Planning Time: 396.548 ms
21	Execution Time: 7672.254 ms

Времето изминато за извршување на операцијата **insert** изнесува :

```
INSERT INTO Reservation
(GuestUseruserId,CampLocationca
mplocationId,number_of_guests,
reservation_status, check_in_date,
check_out_date,total_price)
VALUES (5, 10, 2,
'confirmed',
'2026-07-01',
'2026-07-05', 200.00);
```

ABC QUERY PLAN	
1	Insert on reservation (cost=0.00..0.02 rows=0 width=0) (actual time=10041.913..10041.914 rows=0 loops=1)
2	-> Result (cost=0.00..0.02 rows=1 width=106) (actual time=2.956..2.958 rows=1 loops=1)
3	Planning Time: 0.058 ms
4	Trigger for constraint reservation_guestuserid_fkey: time=0.553 calls=1
5	Trigger for constraint reservation_promotionpromotionid_fkey: time=0.341 calls=1
6	Trigger for constraint reservation_camplocationcamplocationid_fkey: time=0.249 calls=1
7	Execution Time: 10043.092 ms

Времето изминато за извршување на операцијата **update** изнесува :

```
UPDATE Reservation
SET number_of_guests = 3,
total_price = 400.00
WHERE reservationId = 1;
```

ABC QUERY PLAN	
1	Update on reservation (cost=0.42..8.44 rows=0 width=0) (actual time=711.067..711.069 rows=0 loops=1)
2	-> Index Scan using reservation_pkey on reservation (cost=0.42..8.44 rows=1 width=26) (actual time=0.118..0.125 rows=1 loops=1)
3	Index Cond: (reservationid = 1)
4	Planning Time: 0.135 ms
5	Execution Time: 711.111 ms

5. Времето на извршување со индекси изнесува

```
CREATE INDEX idx_reservation_camp_id
ON Reservation
(CampLocationcamplocationId,reservationId);
```

ABC QUERY PLAN	
1	GroupAggregate (cost=29.39..29.56 rows=1 width=141) (actual time=0.261..0.263 rows=1 loops=1)
2	-> Sort (cost=29.39..29.43 rows=14 width=73) (actual time=0.239..0.241 rows=8 loops=1)
3	Sort Key: r.reservationid
4	Sort Method: quicksort Memory: 25kB
5	-> Nested Loop Left Join (cost=5.02..29.13 rows=14 width=73) (actual time=0.219..0.227 rows=8 loops=1)
6	-> Nested Loop Left Join (cost=4.60..20.39 rows=2 width=69) (actual time=0.128..0.130 rows=1 loops=1)
7	-> Index Scan using camplocation_pkey on camplocation cl (cost=0.29..8.31 rows=1 width=61) (actual time=0.080..0.081 rows=1 loops=1)
8	Index Cond: (camplocationid = 100)
9	-> Bitmap Heap Scan on review rv (cost=4.31..12.06 rows=2 width=12) (actual time=0.044..0.044 rows=0 loops=1)
10	Recheck Cond: (camplocationcamplocationid = 100)
11	-> Bitmap Index Scan on idx_review_camp_date (cost=0.00..4.31 rows=2 width=0) (actual time=0.038..0.038 rows=0 loops=1)
12	Index Cond: (camplocationcamplocationid = 100)
13	-> Materialize (cost=0.42..8.58 rows=7 width=8) (actual time=0.089..0.094 rows=8 loops=1)
14	-> Index Only Scan using idx_reservation_camp_id on reservation r (cost=0.42..8.54 rows=7 width=8) (actual time=0.084..0.087 rows=8 loops=1)
15	Index Cond: (camplocationcamplocationid = 100)
16	Heap Fetches: 0
17	Planning Time: 0.856 ms
18	Execution Time: 0.367 ms

6. Времето на извршување на операцијата *Insert по индексирање* изнесува

```
INSERT INTO Reservation
(GuestUseruserId,CampLocationca
mplocationId,number_of_guests,
reservation_status, check_in_date,
check_out_date,total_price)

VALUES (5,
10,
2,
'confirmed',
'2026-07-01',
'2026-07-05', 200.00);
```

	ABC QUERY PLAN
1	Insert on reservation (cost=0.00..0.02 rows=0 width=0) (actual time=71.089..71.090 rows=0 loops=1)
2	-> Result (cost=0.00..0.02 rows=1 width=106) (actual time=1.850..1.851 rows=1 loops=1)
3	Planning Time: 0.106 ms
4	Trigger for constraint reservation_guestuserid_fkey: time=52.396 calls=1
5	Trigger for constraint reservation_promotionpromotionid_fkey: time=0.163 calls=1
6	Trigger for constraint reservation_camplocationcamplocationid_fkey: time=28.981 calls=1
7	Execution Time: 152.664 ms

Времето на извршување на операцијата **update по индексирање** изнесува

```
UPDATE Reservation
SET number_of_guests = 5,
total_price = 600.00
WHERE reservationId = 1;
```

	ABC QUERY PLAN
1	Update on reservation (cost=0.42..8.44 rows=0 width=0) (actual time=0.867..0.867 rows=0 loops=1)
2	-> Index Scan using reservation_pkey on reservation (cost=0.42..8.44 rows=1 width=26) (actual time=0.593..0.602 rows=1 loops=1)
3	Index Cond: (reservationid = 1)
4	Planning Time: 0.149 ms
5	Execution Time: 10.599 ms