ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«САНКТ-ПЕТЕРБУРГСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ПЕТРА ВЕЛИКОГО»
Высшая школа программной инженерии

ОТЧЕТ ПО ПРАКТИЧЕСКИМ ЗАДАНИЯМ
по дисциплине «Базы данных»

Студент
3530202/90202   Потапова А. М.

Преподаватель   Прокофьев О. В.

Санкт-Петербург
2022 г.

Содержание

Практическое задание №3.1 Проектирование аналитической схемы базы данных.

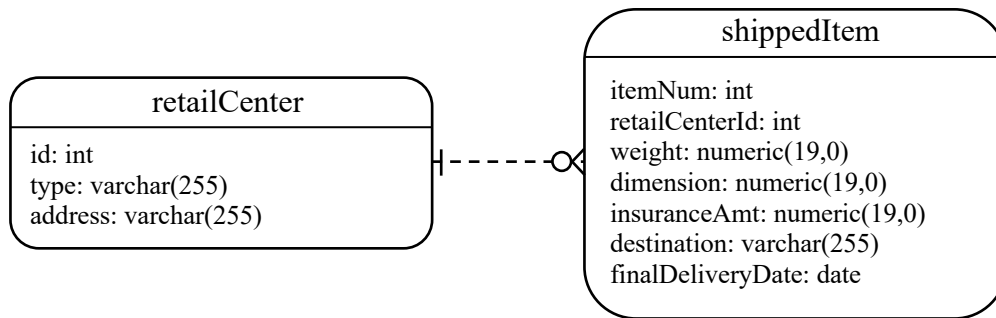*Рисунок 1. ER-диаграмма*

```sql
CREATE TABLE retailCenter (
    id          int primary key generated always as identity,
    type        varchar(255),
    address     varchar(255)
);

CREATE TABLE shippedItem (
    itemNum             int primary key generated always as identity,
    retailCenterId      int references retailCenter (id),
    weight              numeric(19, 0),
    dimension           numeric(19, 0),
    insuranceAmt        numeric(19, 0),
    destination         varchar(255),
    finalyDeliveryDate  date
);

CREATE OR REPLACE PROCEDURE insertDataToRetailCenter()
LANGUAGE plpython3u
as $$
    import random

    types = ["Corner Store", "Convenience center", "Neighborhood center", "Community center", "Lifestyle center"]

    addresses = ["Lindsay Common", "293 Konopelski Circles Suite 934", "Hilton Green", "1940 Antoinette Pine Apt. 729", "Nolan Point", "674 Cormier Coves", "Sauer Mount Apt. 315", "659 Waelchi Wall", "Arlo Square Apt. 533", "7300 Rhea Burgs", "Wolf Trafficway Suite 688", "439 Rogelio Extension Apt. 541", "770 Harmony Skyway Apt. 577", "Haag Harbors Apt. 353", "1183 Ali Street", "Berge Forges", "Cassin Mall Suite 964", "Morissette Springs Suite 212", "556 Mayert Rest Apt. 789", "600 Lebsack Villages", "6743 Jenkins Camp Apt. 289", "688 Bergnaum Mews", "2605 Dangelo Ridges", "944 Hermiston Isle Apt. 228", "313 Conner Garden", "4957 Sanford Greens Suite 075", "Brown Mountains Apt. 210", "Samantha Inlet Suite 678", "758 Tierra Center", "4135 Ansel Well", "Werner Trace Apt. 718", "Katlynn Unions Suite 306", "102 Wilderman Common", "Tyrique Harbors Suite 211", "793 Upton Walk", "Jacobson Club Suite 672", "Kohler Plains Suite 892", "Maryjane Valleys Suite 960", "38 Lester Walks", "Ullrich Islands Suite 581", "Audra Motorway Apt. 035", "Runte Divide Suite 759", "610 Smith Prairie", "Legros Knoll Apt. 741", "Greenholt Fords Suite 978", "Conn Forges Apt. 901", "927 Brad Square Apt. 988", "43 Jast Springs Suite 930", "411 Christa Mount Suite 224", "469 Darius Bridge", "3701 Liam Wall Suite 101", "Kerluke Rapids Apt. 099", "281 Juliet Mountain", "13 Adah Forks Apt. 777", "Rice Ridge Suite 028"]
    cities = ["Grand Ranportvey", "Saint Hunmontwest", "Bampstianperley", "Great Buckthbay", "Werssle", "Blobourgcombe", "Phistho Ferry", "West Johnsbramp", "West Lkshamstairs", "Hexcor", "Mieka Bay", "Noaksnte", "Gasbrackpool", "Nowhea" Innesvil", "Camlein Sands", "Lothmas-In-Stryulnorth", "West Minirthllou", "Caveren", "New Innesvil", "Camlein Sands", "Lothmas-In-Stryulnorth", "West Minirthllou", "Caveren", "Wyongwimfallsford", "Port Havresou", "Midyorkfield", "Grand Wisjust", "Red Wanwar With Tagrays", "Ntunalta", "Chelmeau", "Beachkin", "Fronstone", "Port Linor Du Abdge", "Grand Dsor","West Woodnace Under Mature", "Whadock Upon Cultgrave End", "Riordesdoches", "Port Ashphurwem",
```

```
"South Kuujrough", "Yelteley", "Lake Sheywe", "Cotesgladehove", "Pordor Aux Sworthannesham",
"Miawnaarms", "Red Smyto", "Lishitcombe", "Port Phamkeynston", "Rkby Lake"]

   file = open("retail.csv", "w")

   for i in range(1000001) :
      type = types[random.randint(0, len(types) - 1)]
      address = cities[random.randint(0, len(cities)-1)] + ' ' + addresses[random.randint(0, len(addresses)-1)]
      line = type + ',' + address + '\n'
      file.writelines(line)

   file.close()
$$;

CALL insertDataToRetailCenter();
COPY retailCenter (type, address) FROM 'retail.csv' CSV HEADER DELIMITER ',';

CREATE OR REPLACE PROCEDURE insertDataToShippedItem()
language plpython3u
as $$
   import random
   import datetime

   address = ["Blick Trafficway Apt. 254", "8 Walter Parkways Suite 036", "33990 Deron Mews", "783
Bernhard Course", "Predovic Hollow Suite 619", "479 Anthony Course", "9487 Afton Orchard", "97 Rubie
Point", "4136 Estevan Burg", "763 Collier Harbor", "Flatley Manor Apt. 965", "356 Susan Ports", "9 Becker
Viaduct", "813 Labadie Forks", "885 Schmeler Plaza", "36 Bogan Plains", "Bosco Lane Suite 424", "802
Morar Meadows", "44 Demetrius Court", "Brendan Vista Apt. 127", "Morissette Creek Suite 469", "163
Kirsten Ways", "Hermiston Light Apt. 710", "7241 Faustino Avenue", "161 Gerard Path", "958 Ankunding
Turnpike", "7239 Ashleigh Port", "99 Schultz Squares", "Roberts Island Suite 238", "Lind Trafficway Suite
476", "566 Tessie Shore", "Willms Harbor Suite 040", "Gerlach Views Suite 759", "Bobbie Wells Apt.
409", "238 Kuphal Parks Apt. 494", "Kris Mission Suite 291", "88 Treutel Port Apt. 913", "87 Evert
Overpass", "4 Crist Green", "806 Eugene Lodge Apt. 319", "5756 Delfina Harbor", "399 Nienow Rapid",
"Romaguera Hollow Apt. 905", "284 Leone Valleys", "Fred Mission Apt. 143", "90 Bartell Parkways", "Orn
Streets Suite 969", "Reta Underpass Apt. 098", "849 Rickey Burgs", "1310 Emerald Dam", "Klocko Road
Suite 012", "Wisozk Junctions Apt. 664", "Loraine Shoal Suite 891", "658 Bode Tunnel", "Langworth Isle
Apt. 557", "7 Moore Mountain Apt. 734", "568 Hilpert Squares", "134 Filomena Place", "Monroe Club Apt.
099", "262 Vella Crossroad"]
   cities = ["Ntonbawles", "Barreing", "San Conthill", "Great Kaldenkur", "Rseychard", "Folkkit", "West
Hampmeau", "North Mount", "Fort Thorpemidlod", "Capetown", "East Francesbotlin", "Cidrid", "Raistree
Hills", "Port Moslage", "Toonmod", "Port Terreplaines", "North Hirith", "Port Dealheadcarn", "East
Phoeristid", "Grand Havrebay", "Los", "Boncoal", "Greensidekirksea", "Stonemagwer", "South Lomrell",
"Ponmsey River", "Lauwick", "Grand Perthbad", "Santa Skavan", "Reeskebury", "Stonerabrier", "Wicy",
"Lensgasp", "Port Beaubatt", "Lytvi", "Iboylu", "Thensnahythe", "Liammeton", "Pockarms", "Tainster",
"Hostot Harbour", "La Tophia", "Crayworth", "Annboumo"]
   file = open("shipped.csv", "w")

   for i in range(0, 9000001) :
      centerId = random.randint(1, 1000000)
      weight = random.getrandbits(63)
      dimension = random.getrandbits(63)
      insuranceAmt = random.getrandbits(63)
      destination = cities[random.randint(0, len(cities)-1)] + ' ' + address[random.randint(0, len(address)-1)]
      finalyDeliveryDate = datetime.date.today() + datetime.timedelta(random.randint(1, 64))
      line = str(centerId) + ',' + str(weight) + ',' + str(dimension) + ',' + str(insuranceAmt) + ',' + destination +
',' + str(finalyDeliveryDate) + '\n'
      file.writelines(line)
   file.close()
$$;

CALL insertDataToShippedItem();
COPY shippedItem (retailCenterId, weight, dimension, insuranceAmt, destination, finalyDeliveryDate)
FROM 'shipped.csv' CSV HEADER DELIMITER ',';
```

Запрос А

```
SELECT * FROM retailcenter
WHERE address ilike '%' || 'Apt' || '%' AND type = 'Convenience center';
```

Explain Analyze (без индекса)

```
Gather  (cost=1000.00..23896.80 rows=56548 width=57) (actual time=1.395..631.655
rows=61883 loops=1)
  Workers Planned: 2
  Workers Launched: 2
  -> Parallel Seq Scan on retailcenter  (cost=0.00..17242.00 rows=23562 width=57) (actual
time=0.117..596.437 rows=20628 loops=3)
      Filter: (((address)::text ~~* '%Apt%'::text) AND ((type)::text = 'Convenience
center'::text))
      Rows Removed by Filter: 312706
Planning Time: 0.998 ms
Execution Time: 634.556 ms
```

Explain Analyze (с индексом)

```
Bitmap Heap Scan on retailcenter  (cost=2288.31..16291.82 rows=56548 width=57)
(actual time=15.936..243.703 rows=61883 loops=1)
  Recheck Cond: ((type)::text = 'Convenience center'::text)
  Filter: ((address)::text ~~* '%Apt%'::text)
  Rows Removed by Filter: 138617
  Heap Blocks: exact=10992
  -> Bitmap Index Scan on at_indx_retail  (cost=0.00..2274.18 rows=200767 width=0)
(actual time=14.198..14.198 rows=200500 loops=1)
      Index Cond: ((type)::text = 'Convenience center'::text)
Planning Time: 1.330 ms
Execution Time: 246.357 ms
```

Запрос Б

```
SELECT * FROM retailCenter
JOIN shippedItem ON retailCenter.id = shippedItem.retailCenterId
WHERE retailCenter.type = 'Lifestyle center' AND shippedItem.destination ilike '%' ||
'capetown' || '%';
```

Explain Analyze (без индекса)

```
Gather  (cost=19121.32..200320.43 rows=35598 width=137) (actual
time=6063.044..6163.706 rows=40708 loops=1)
  Workers Planned: 2
  Workers Launched: 2
```

```
     -> Parallel Hash Join  (cost=18121.32..195760.63 rows=14832 width=137) (actual
time=6035.104..6082.905 rows=13569 loops=3)
        Hash Cond: (shippeditem.retailcenterid = retailcenter.id)
        -> Parallel Seq Scan on shippeditem  (cost=0.00..174655.00 rows=74785 width=80)
(actual time=0.229..5819.127 rows=67969 loops=3)
           Filter: ((destination)::text ~~* '%capetown%'::text)
           Rows Removed by Filter: 2932031
        -> Parallel Hash  (cost=16200.33..16200.33 rows=82639 width=57) (actual
time=137.517..137.521 rows=66904 loops=3)
           Buckets: 65536  Batches: 8  Memory Usage: 2848kB
           -> Parallel Seq Scan on retailcenter  (cost=0.00..16200.33 rows=82639 width=57)
(actual time=19.251..110.003 rows=66904 loops=3)
              Filter: ((type)::text = 'Lifestyle center'::text)
              Rows Removed by Filter: 266429
Planning Time: 1.882 ms
JIT:
  Functions: 42
  Options: Inlining false, Optimization false, Expressions true, Deforming true
  Timing: Generation 2.824 ms, Inlining 0.000 ms, Optimization 1.800 ms, Emission
56.280 ms, Total 60.904 ms
Execution Time: 6209.904 ms
```

Explain Analyze (с индексом)

```
Gather  (cost=19121.32..200320.43 rows=35598 width=137) (actual
time=5400.888..5506.240 rows=40708 loops=1)
  Workers Planned: 2
  Workers Launched: 2
  -> Parallel Hash Join  (cost=18121.32..195760.63 rows=14832 width=137) (actual
time=5375.993..5412.803 rows=13569 loops=3)
     Hash Cond: (shippeditem.retailcenterid = retailcenter.id)
     -> Parallel Seq Scan on shippeditem  (cost=0.00..174655.00 rows=74785 width=80)
(actual time=0.166..5141.555 rows=67969 loops=3)
        Filter: ((destination)::text ~~* '%capetown%'::text)
        Rows Removed by Filter: 2932031
     -> Parallel Hash  (cost=16200.33..16200.33 rows=82639 width=57) (actual
time=181.136..181.138 rows=66904 loops=3)
        Buckets: 65536  Batches: 8  Memory Usage: 2848kB
        -> Parallel Seq Scan on retailcenter  (cost=0.00..16200.33 rows=82639 width=57)
(actual time=38.598..136.762 rows=66904 loops=3)
           Filter: ((type)::text = 'Lifestyle center'::text)
           Rows Removed by Filter: 266429
Planning Time: 3.614 ms
JIT:
  Functions: 42
  Options: Inlining false, Optimization false, Expressions true, Deforming true
  Timing: Generation 6.211 ms, Inlining 0.000 ms, Optimization 4.181 ms, Emission
111.796 ms, Total 122.188 ms
Execution Time: 5798.198 ms
```

Запрос B

```
SELECT address FROM retailCenter
WHERE to_tsvector('english', address) @@ to_tsquery('north');
```

Explain Analyze (без индексов)

```
Gather  (cost=1000.00..226033.67 rows=5000 width=36) (actual time=5023.417..5036.780
rows=0 loops=1)
  Workers Planned: 2
  Workers Launched: 2
  -> Parallel Seq Scan on retailcenter  (cost=0.00..224533.67 rows=2083 width=36) (actual
time=4984.836..4984.839 rows=0 loops=3)
      Filter: (to_tsvector('english'::regconfig, (address)::text) @@ to_tsquery('north'::text))
      Rows Removed by Filter: 333333
Planning Time: 3.014 ms
JIT:
  Functions: 12
  Options: Inlining false, Optimization false, Expressions true, Deforming true
  Timing: Generation 9.735 ms, Inlining 0.000 ms, Optimization 1.136 ms, Emission 27.131
ms, Total 38.002 ms
Execution Time: 5081.038 ms
```

Explain Analyze (с индексом)

```
Bitmap Heap Scan on retailcenter  (cost=67.00..11484.79 rows=5000 width=36) (actual
time=0.053..0.053 rows=0 loops=1)
  Recheck Cond: (to_tsvector('english'::regconfig, (address)::text) @@
to_tsquery('north'::text))
  -> Bitmap Index Scan on gin_idx_retail  (cost=0.00..65.75 rows=5000 width=0) (actual
time=0.048..0.048 rows=0 loops=1)
      Index Cond: (to_tsvector('english'::regconfig, (address)::text) @@
to_tsquery('north'::text))
Planning Time: 3.498 ms
Execution Time: 0.152 ms
```

Практическое задание №3.2 Использование документно-ориентированных объектов типа Json.

Модель не была изменена.

```
SELECT info['primaryName'] AS Name, info['birthYear'] AS BirthDate,
info['deathYear'] AS DeathDate FROM imdb
WHERE info['birthYear'] != '"none"' AND info['deathYear'] != '"none"' AND
info['birthYear'] > '"1900"' AND info['deathYear'] < '"2000"';

SELECT jsonb_array_elements(info['roles']) FROM imdb
WHERE info['primaryName'] = '"Arlette Aarts"';

SELECT * FROM
(
    SELECT
    jsonb_array_elements(info['roles']) AS role_json,
    info['primaryName'] AS primaryName
    FROM imdb
) AS temp
WHERE temp.role_json->>'year' = '1949';

SELECT * FROM imdb
WHERE imdb.info->'roles' @> '[{"year": "2001"}]'::jsonb AND info['roles'] @>
'[{"year": "2000"}]'::jsonb
```
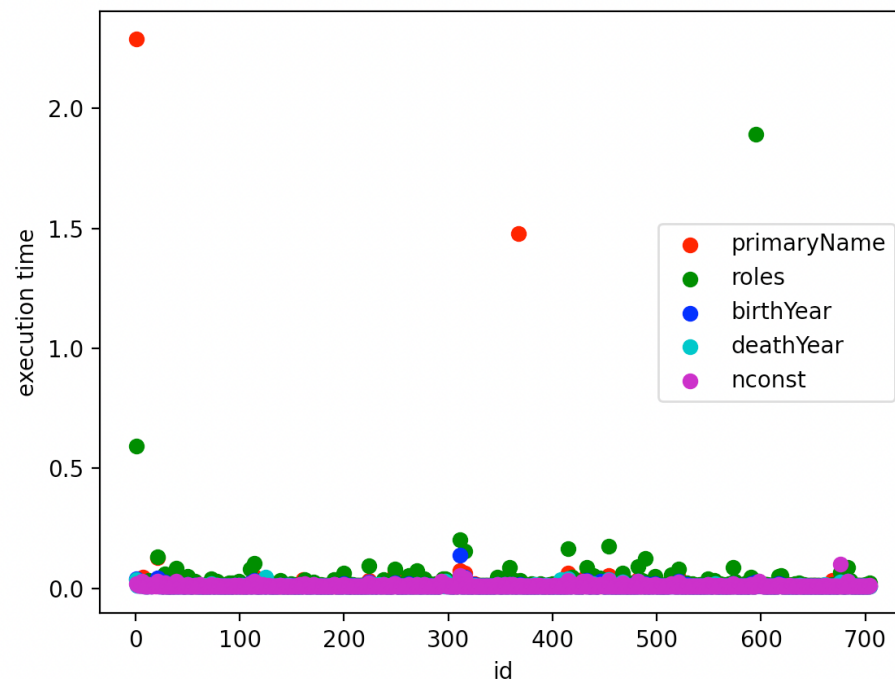

Рисунок 2. График времени исполнения

Размер JSON	10290	186
Размер таблицы ДО	475136	483328
Размер таблицы ПОСЛЕ	483328	483328
Прирост	8192	незначительный
WAL ДО	33554432	33554432
WAL ПОСЛЕ	33554432	33554432

Таблица 1. Таблица изменения размера БД

Вывод

В ходе выполнения данных лабораторных работ был изучен язык plpython3u, парсинг данных и перенос json в базу Postgres. Трудности возникли на этапе установки расширения plpython3u и, поскольку, справится с этой задачей не удалось, было принято решение выполнять работу на приобретенном удаленном сервере. Таким образом, помимо вышеперечисленного, были изучены аспекты работы с PostgreSQL на удаленном сервере.