Саратовский государственный технический университет им.

Гагарина Ю.А.

Институт прикладных информационных технологий и коммуникаций

Кафедра прикладные информационные технологии

Практическая работа №8

Выполнил студент 3 курса

Группы б1ИФСТ-31

Песчанов Арсений Андреевич

Преподаватель: Кузьмин Алексей Константинович

Саратов 2023

****

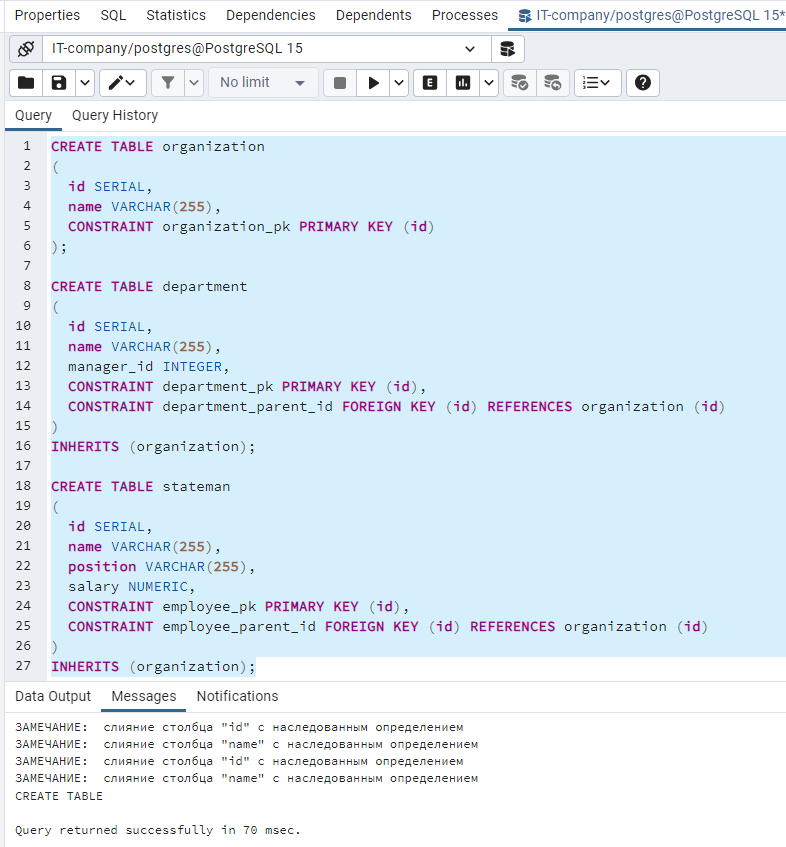
**Занятие 8. ООП**

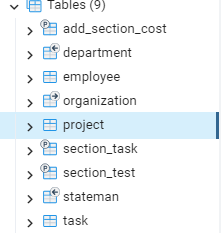
**Практическое задание**

Практическое задание сдается в виде отчета, в котором для каждого задания, подразумевающего практическую работу с экземпляром Postgres, приводятся снимки экрана, на которых видны фрагменты соответствующих экранных форм, комментированные исходные коды запросов, а также результатов их выполнения на экземпляре. Также сдаётся текстовый файл \*.sql, в который включены комментированные исходные коды всех запросов по всему заданию. В случае, если задание требует использования возможности, поддержка которой в Postgres (или используемой версии Postgres) отсутствует, то необходимо отразить это в отчете и использовать альтернативный путь решения задачи.

В качестве «полигона» используйте таблицы (если необходимо), созданные в рамках подготовки ДЗ с одного из предыдущих занятий.

1. Продолжайте продвигаться на SQL-ex.ru.
2. Просмотрите материалы с занятия. Пропустите всё ещё раз через себя. Установите и осознайте принципиальный смысл всех вещей, которые там рассматривались.
3. Реализуйте 3-4 объектных типа, которые находятся между собой в отношениях «предок-потомок». Это может быть, например, один базовый класс с потомками, у некоторых из которых есть свои потомки. Т.е. структура связей должна образовывать некое произвольное дерево.



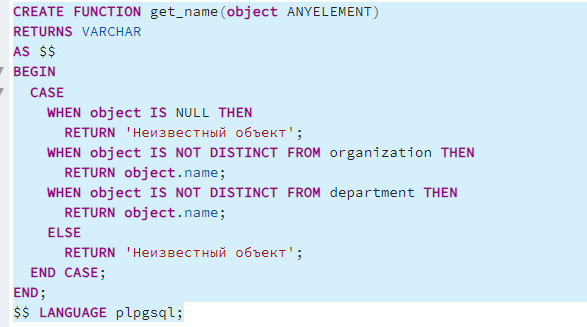


Связь между таблицами будет реализована с помощью столбца parent\_id. Если значение этого столбца равно NULL, то объект является корневым. Если значение этого столбца не равно NULL, то объект является потомком объекта с идентификатором, равным значению столбца parent\_id.

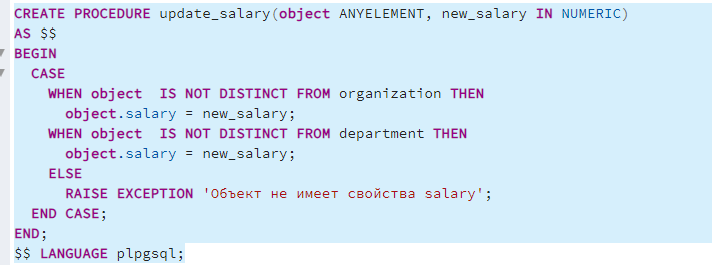
1. Наглядно продемонстрируйте в коде проявление полиморфизма. Как статического (при котором выбор той или иной перегруженной реализации метода может быть произведён на этапе разбора кода), так и динамического (при котором выбор той или иной реализации производится «на лету» в зависимости от того, объект какого из конкретного типа сейчас обрабатывается.

в PostgreSQL нет самого полимормизма, но есть понятие полиформных функций.

Эта функция использует оператор isinstance() для проверки типа объекта. Если объект является типом organization, то функция возвращает значение свойства name объекта. Если объект является типом department, то функция также возвращает значение свойства name объекта. В противном случае функция возвращает строку 'Неизвестный объект'.

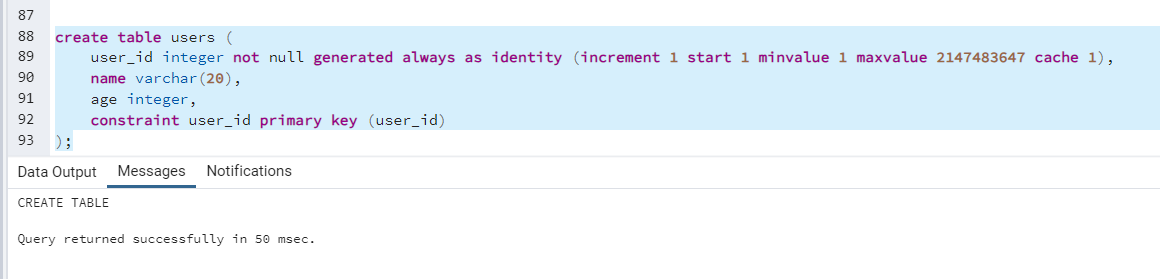


Для демонстрации динамического полиморфизма создадим процедуру update\_salary(), которая будет обновлять зарплату объекта. Процедура будет принимать в качестве аргумента объект любого типа, который имеет свойство salary.

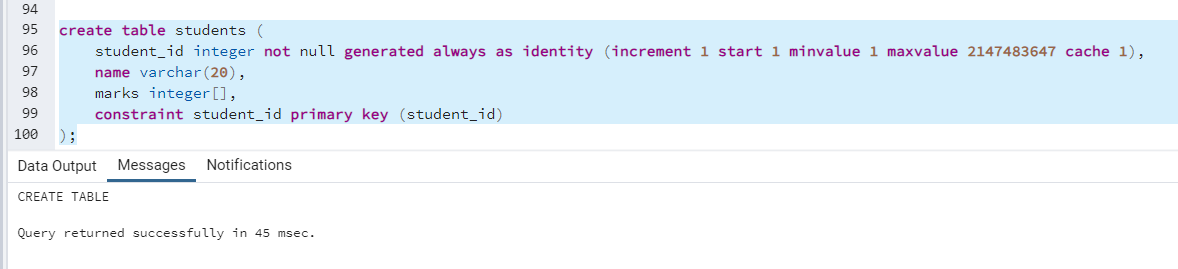


1. Создайте три типа коллекции и реализуйте добавление в них нескольких объектных элементов, итеративный перебор и некую обработку всех элементов, а также очистку коллекции.

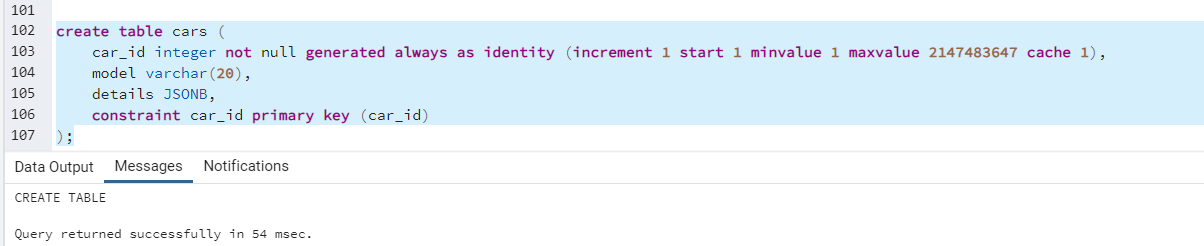
Таблицы



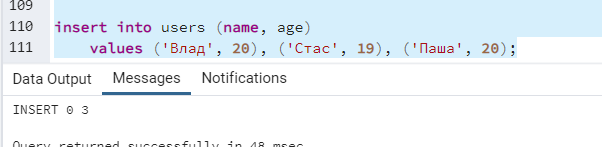
Массивы

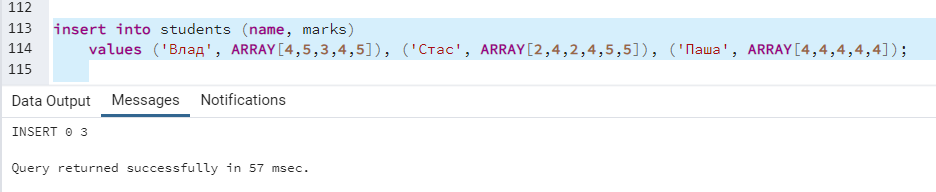


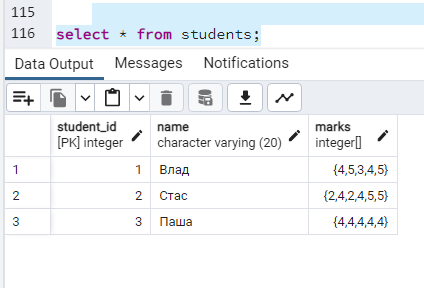
JSON/JSONB

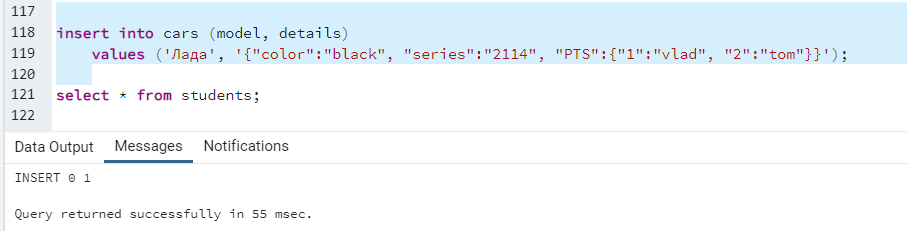


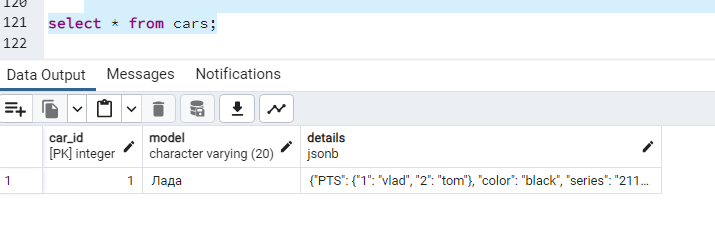
Добавление данных











Итеративный перебор элементов коллекции:



Очистка коллекции

