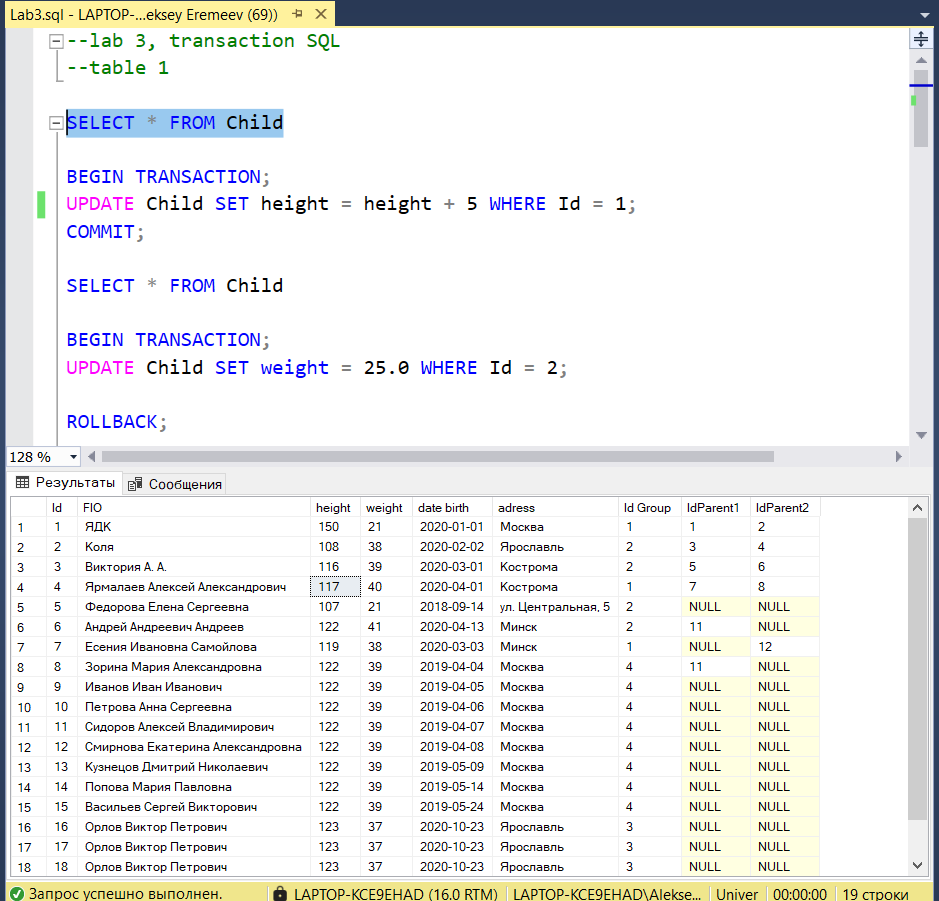
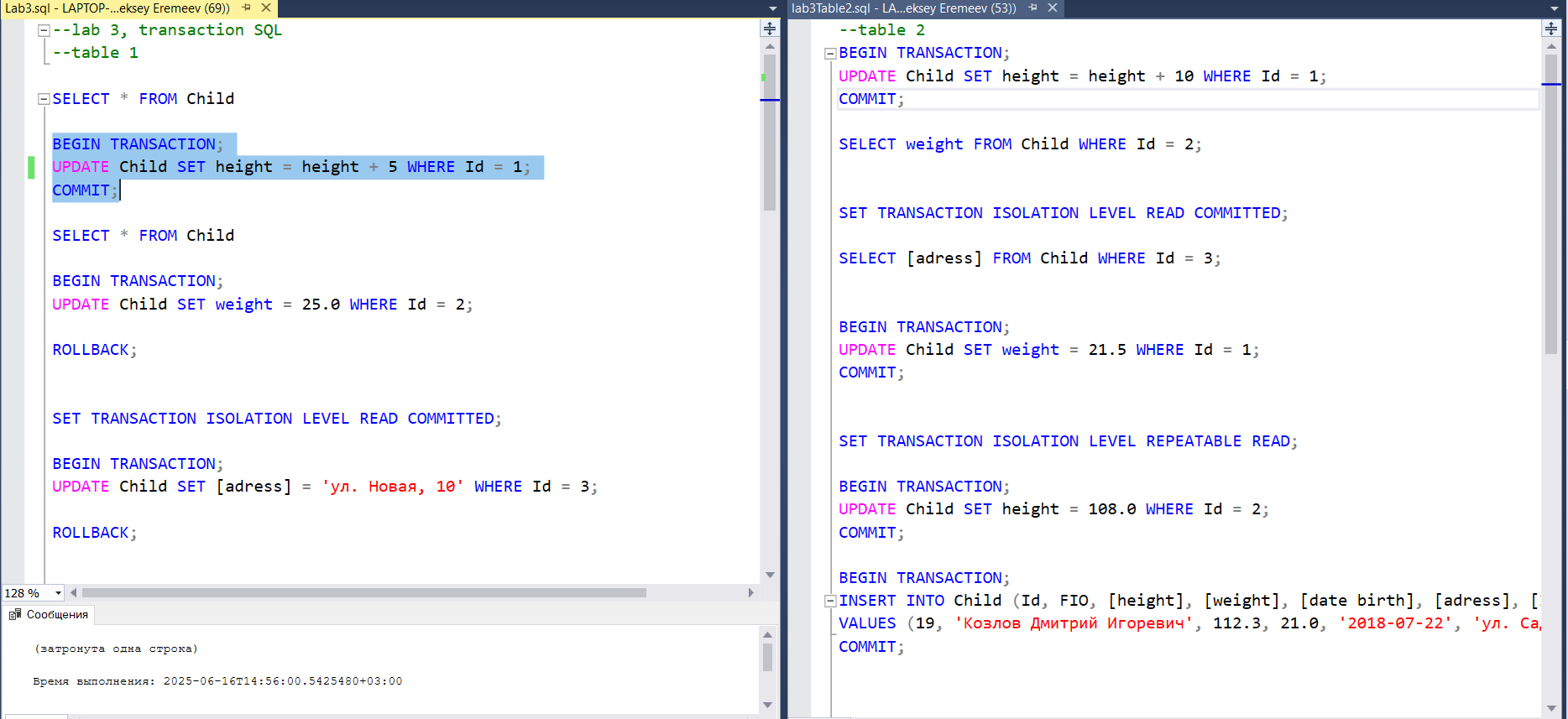
**READ UNCOMMITTED**

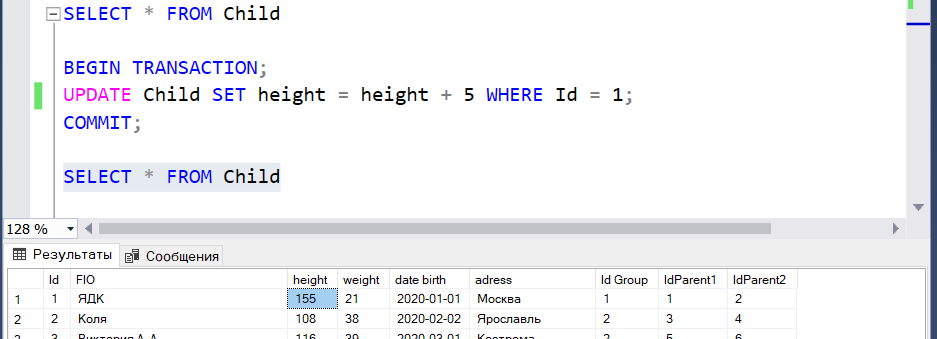


В данном примере до выполнения первой транзакции все данные height не увеличены не на 5 не на 10.

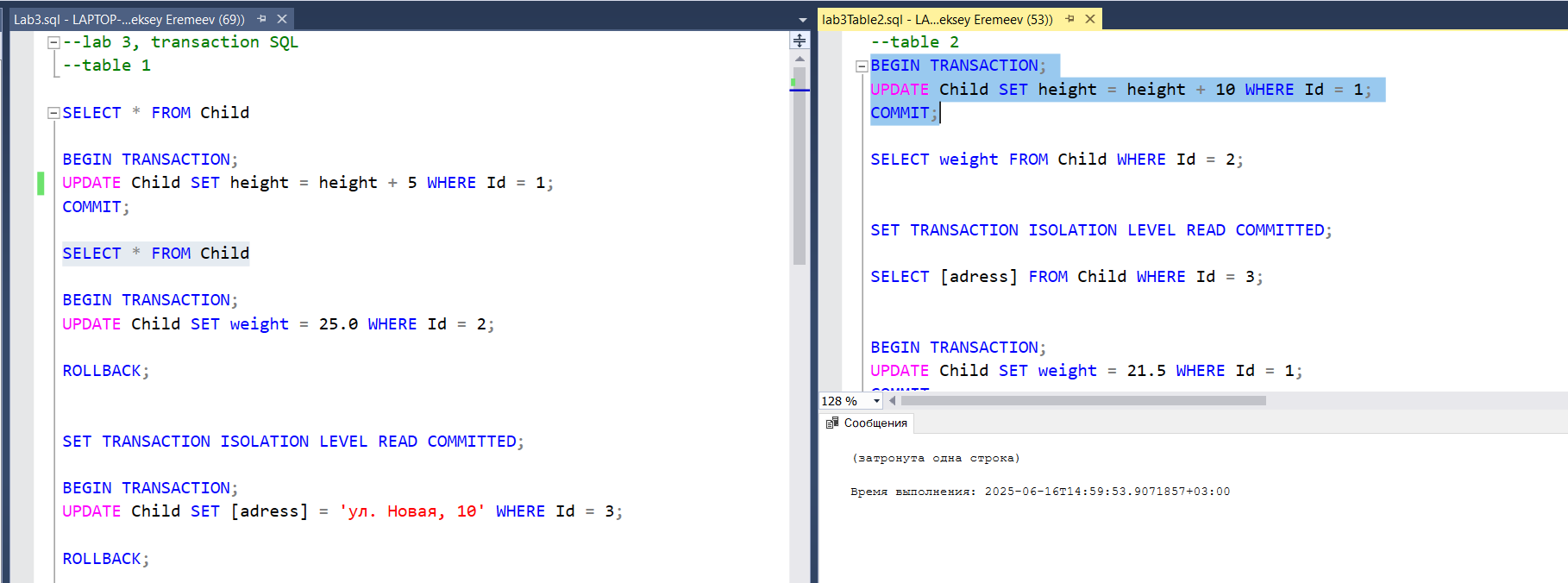
-Выполнение проверки потерянных изменений



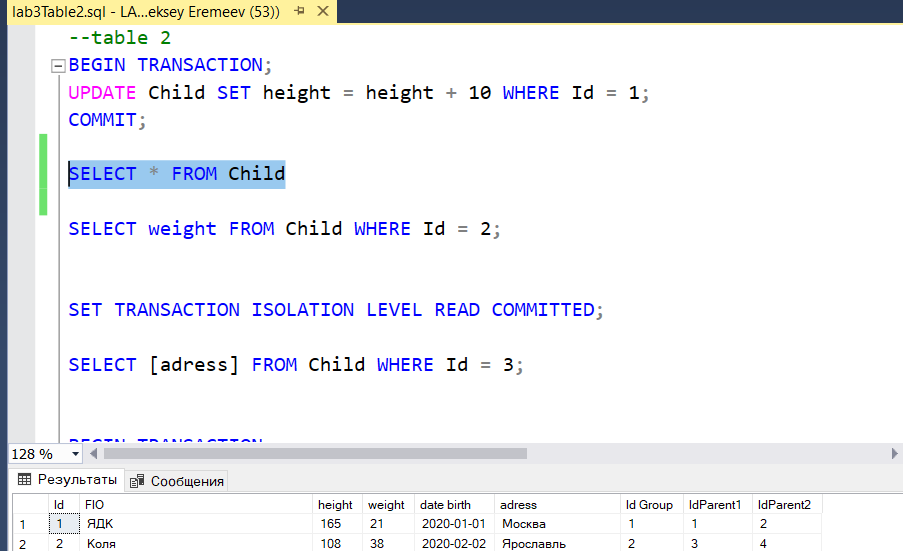
В данном примере выполняется транзакция в первом сеансе. Происходит создание странзакции, увеличение значения height на 5 и фиксация транцакции.



Результат выполнения транзакции.



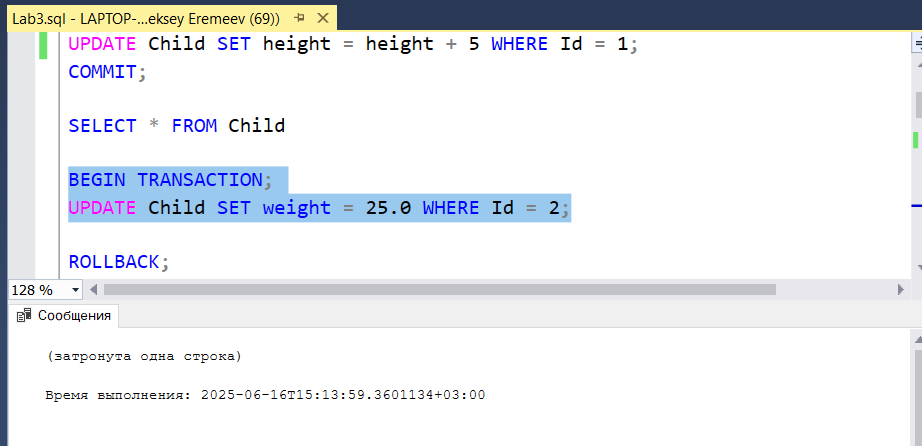
Совершается транзакция второго сеанса. Изменение значения height на 10.



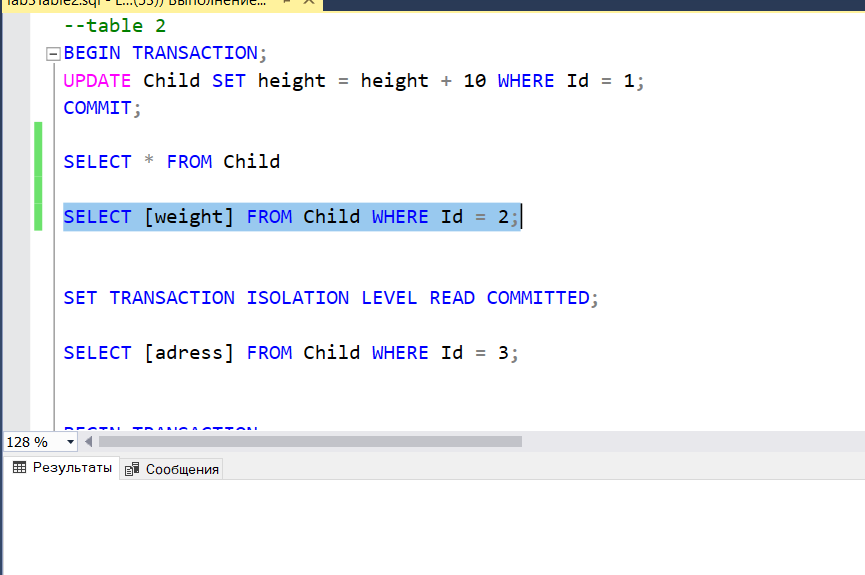


Результат фиксации второй транзакции. Изменение значения на 10, итоговое изменение значения на 15. Нет потерянных изменений, так как каждая транзакция использует последнее зафиксированное изменение.

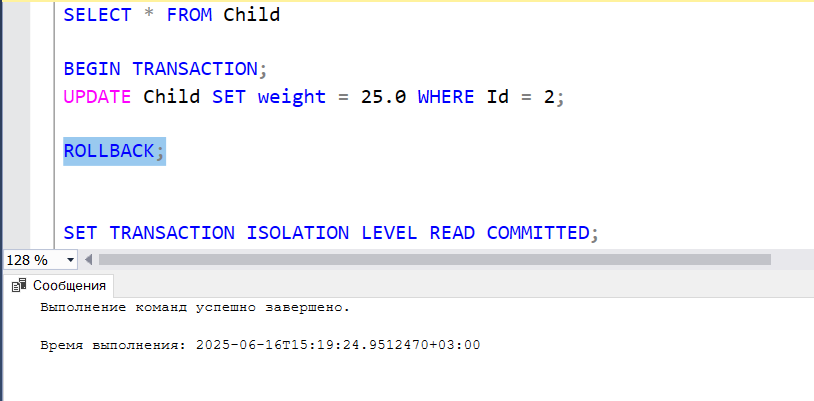
- Выполнение проверки грязного чтения



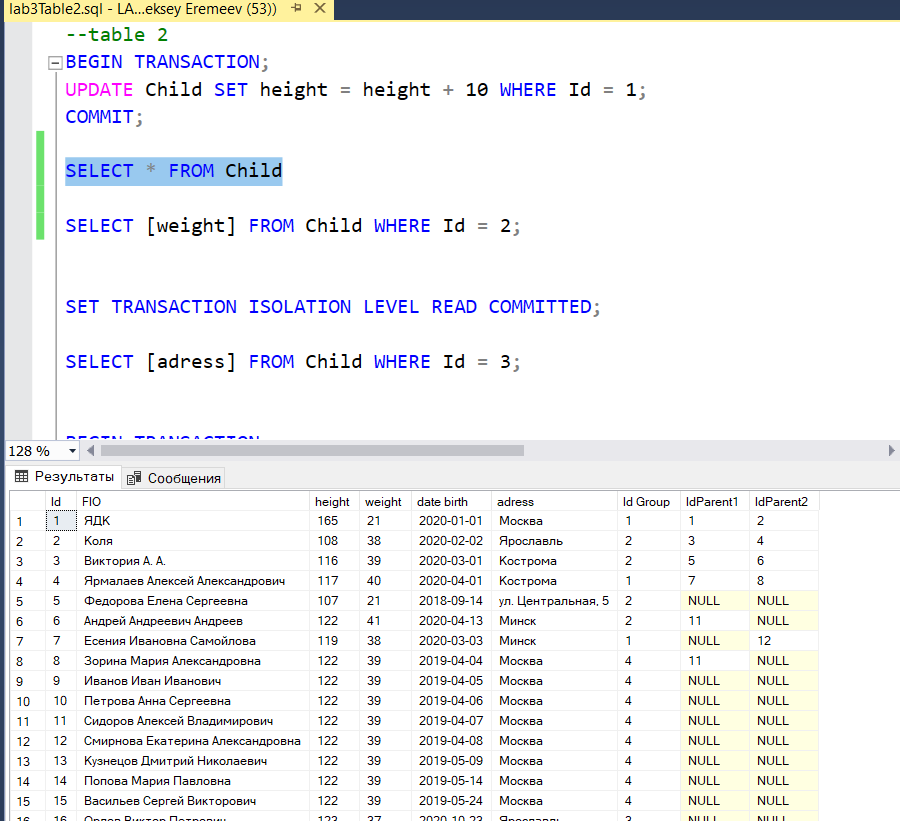
Создается транзакции, изменения значения weight, без фиксации изменения.



Во втором сеансе попытка просмотра данных, сеанс не может выполнить запрос, так как предыдущая транзакция не была зафиксирована.



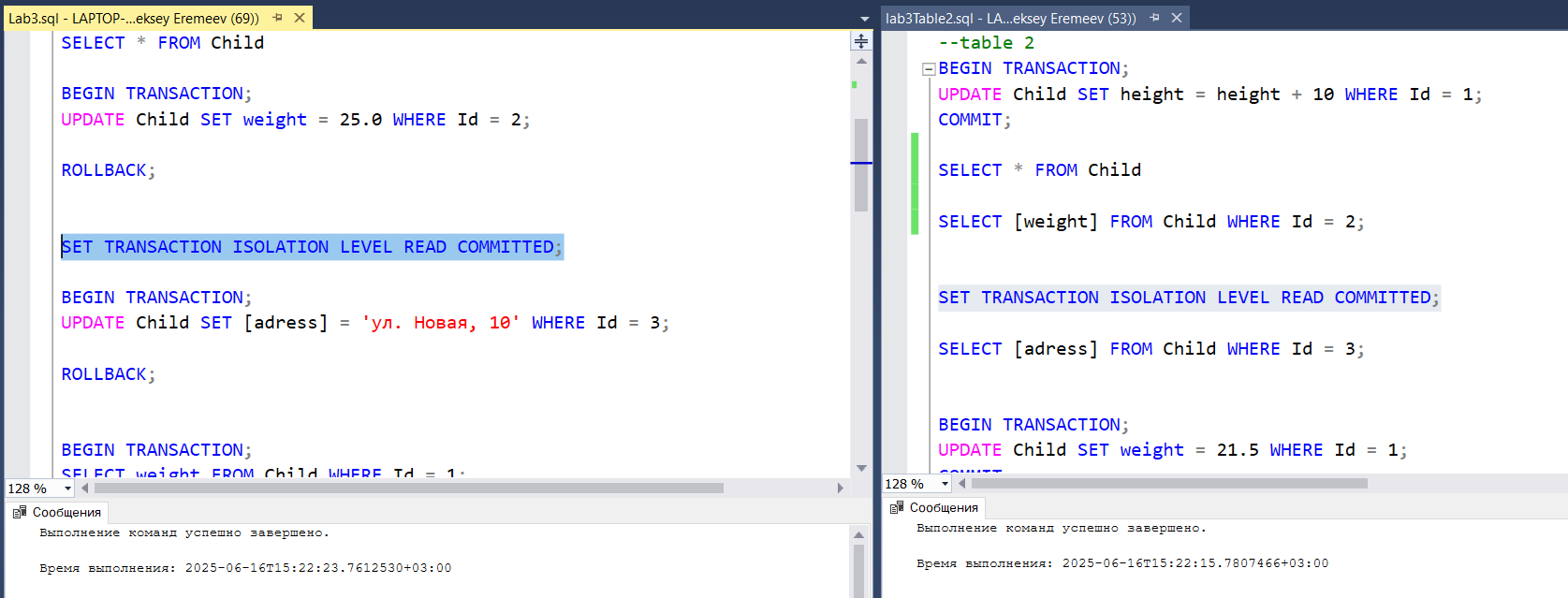
В первом сеансе откатываем изменения.



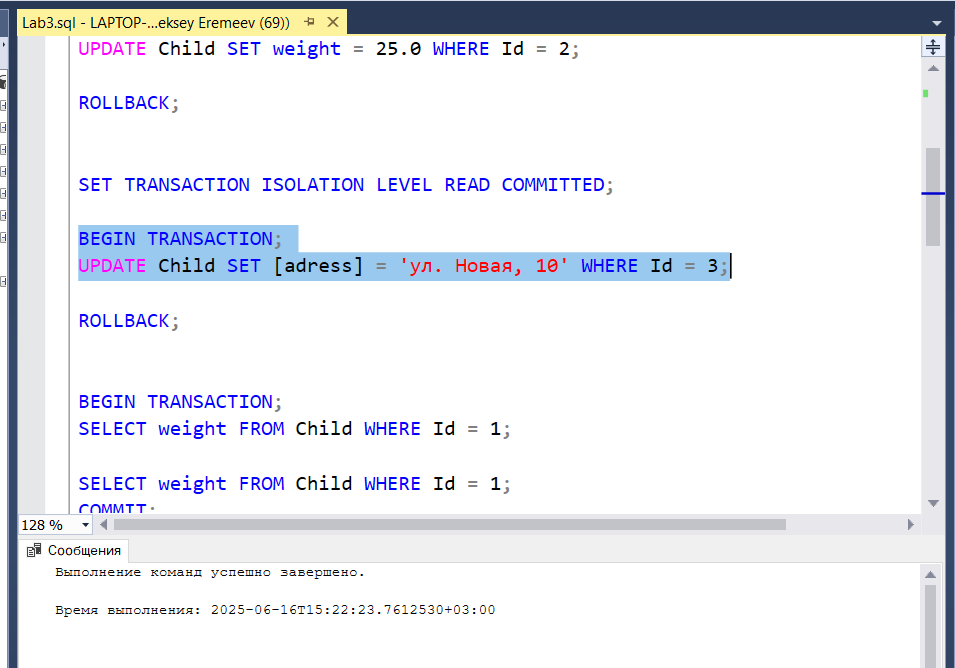
Второй сеанс после фиксации видит данные без изменений транзакций первого сеанса.

**READ COMMITTED**

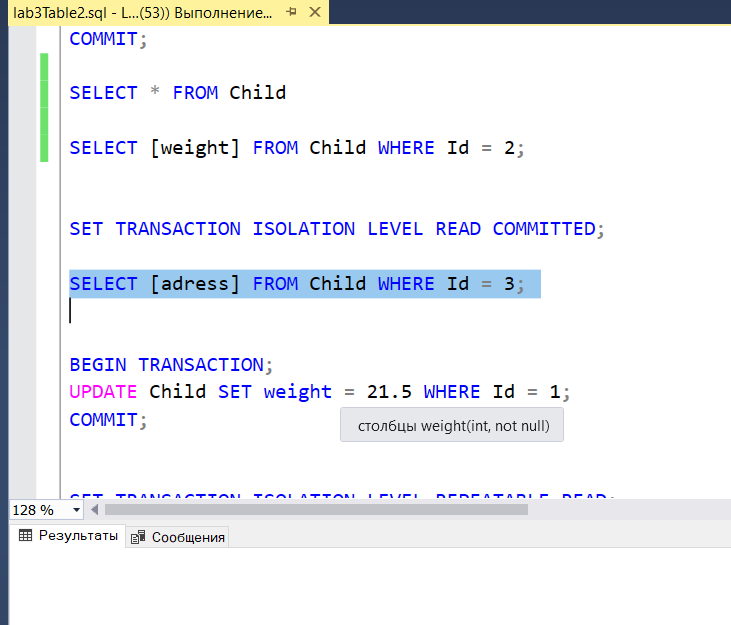
**-** **Проверка грязного чтения**

****

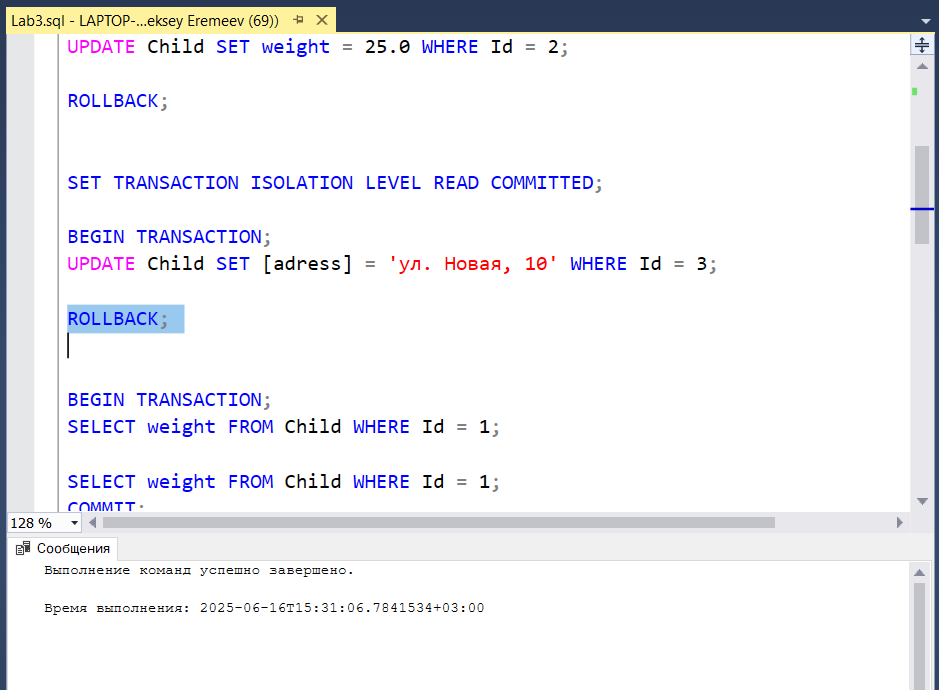
Установка в обоих сеансах уровня READ COMMITTED



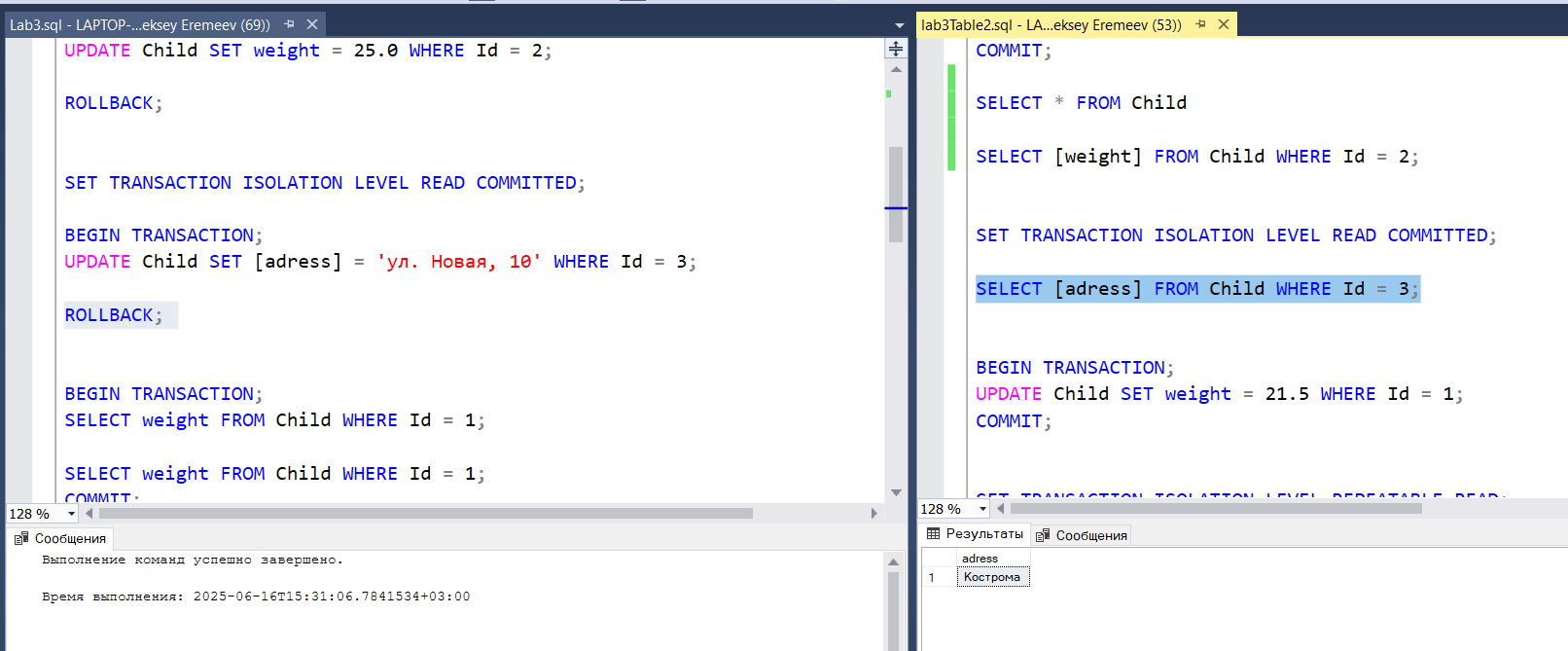
В первом сеансе создаем транзакцию, где изменяем значение address, без фиксации изменений.



Во втором сеансе выполняем выбор данных, но данные не видны, так как изменения не зафиксированы.

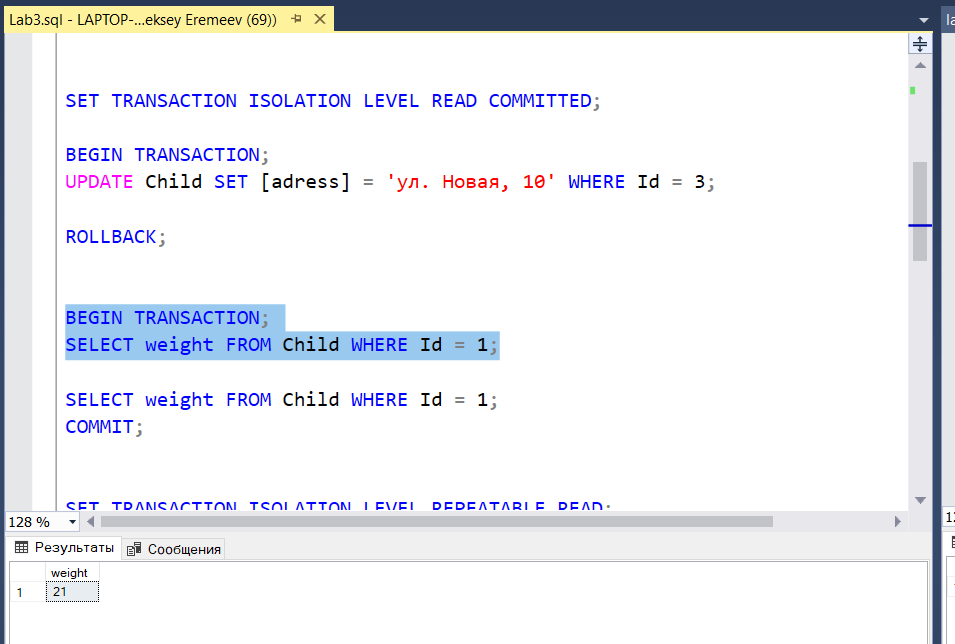


В первом сеансе выполняем откат изменений.

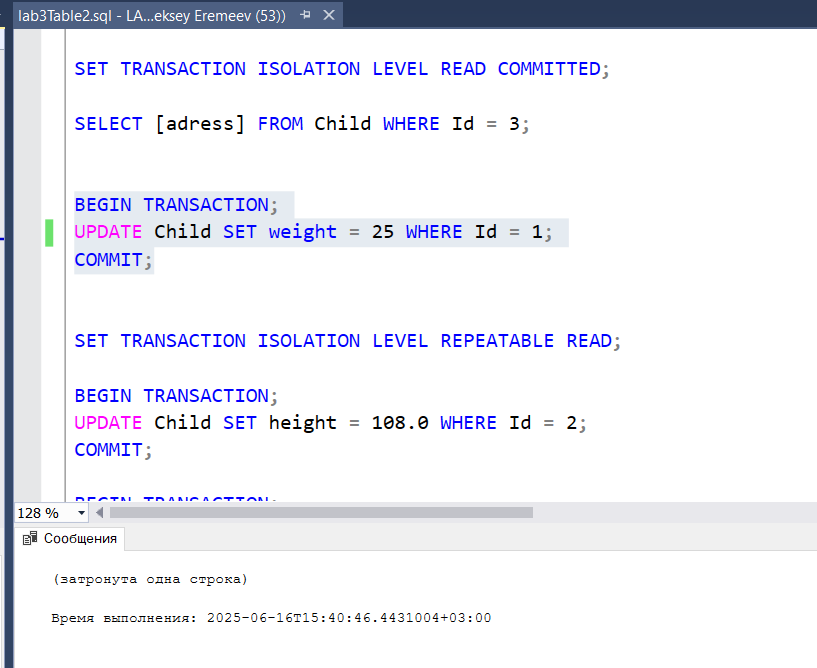


После выполнения отката в первом сеансе, во стром сеансе видны старые данные, без изменения с первого сеанса

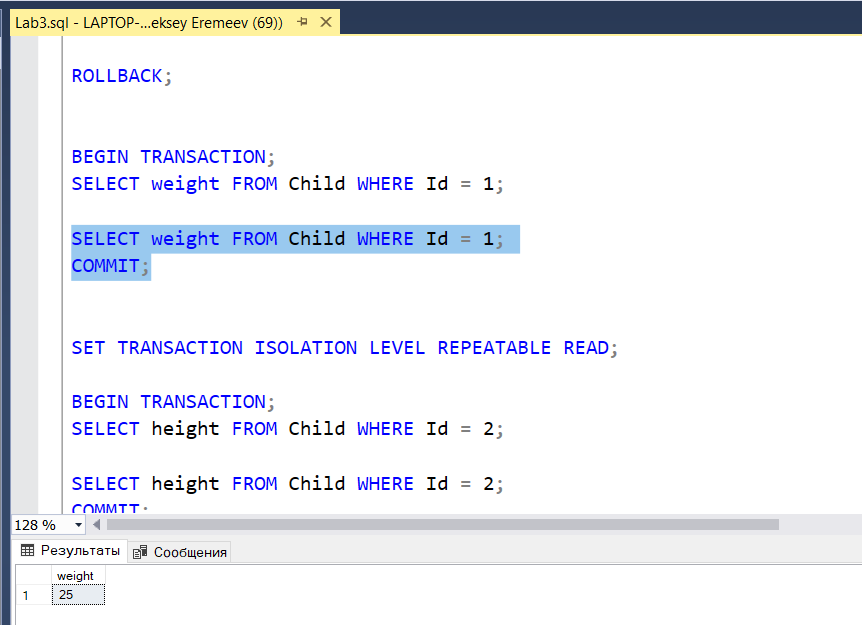
- **Проверка неповторяющегося чтения**



В первом сеансе выполняем создание транзакции чтения, без фиксации.



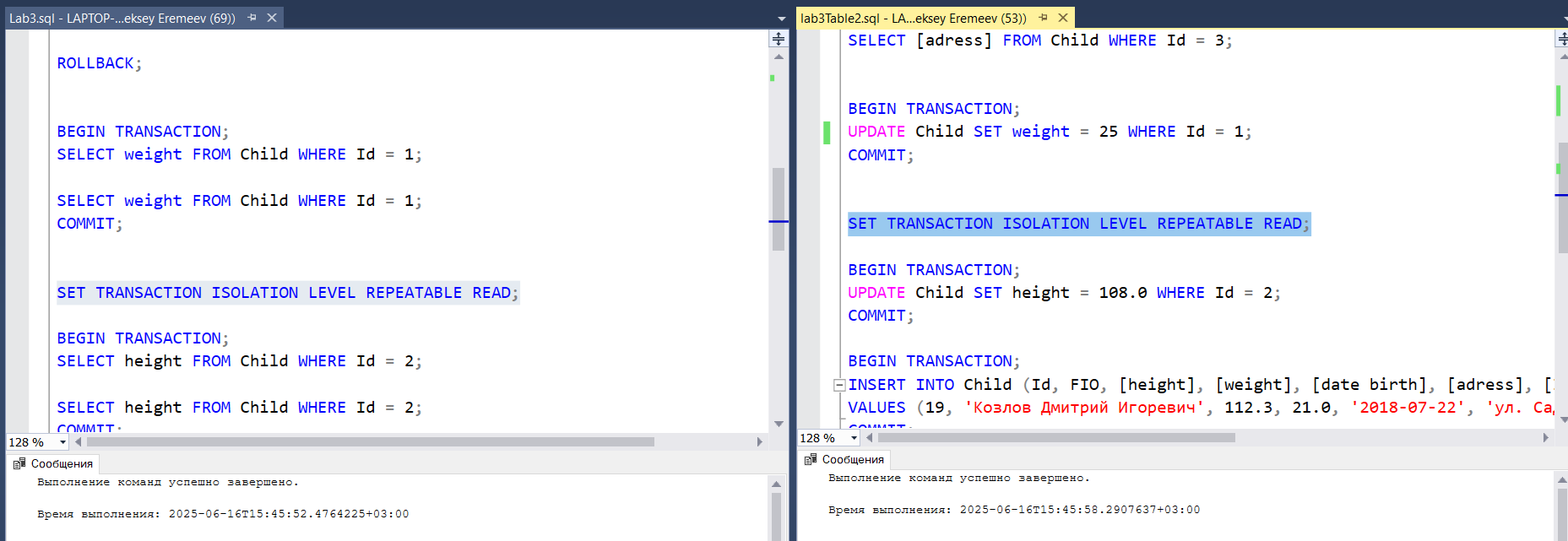
Во втором сеансе создаем транзакцию с изменением данных, и фиксации их.



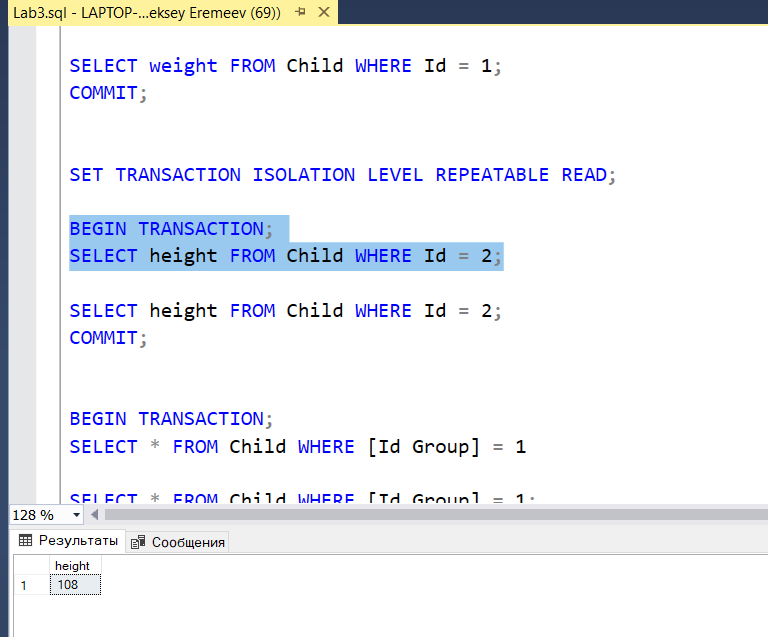
В первом сеансе выполняем второе чтение данных, и фиксацию транзакции, после внесения изменения второго сеанса видим уже измененные данные.

**REPEATABLE READ**

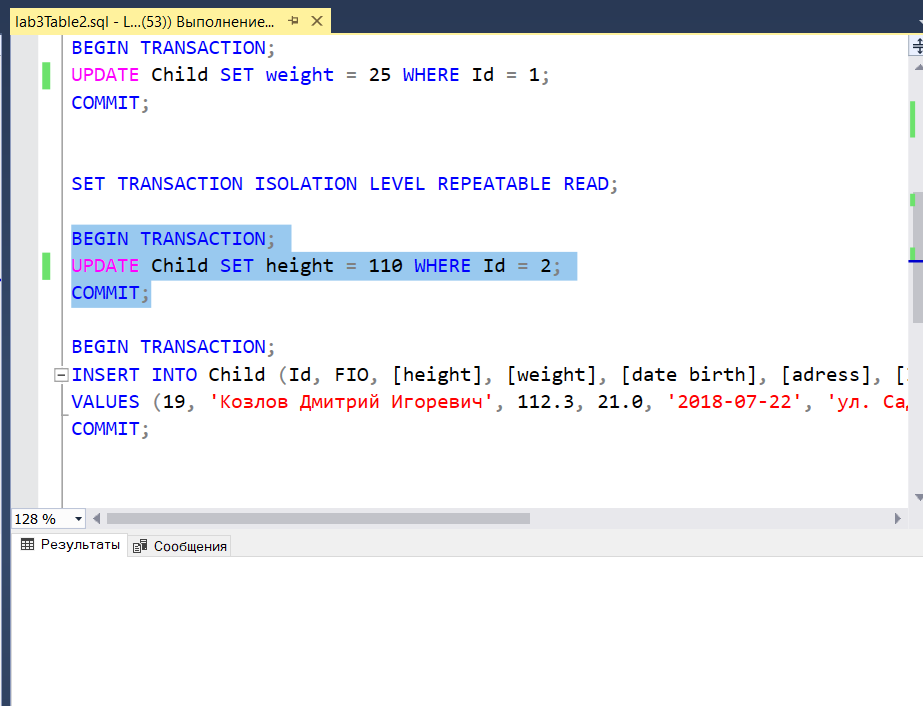
**-** **Проверка неповторяющегося чтения**

****

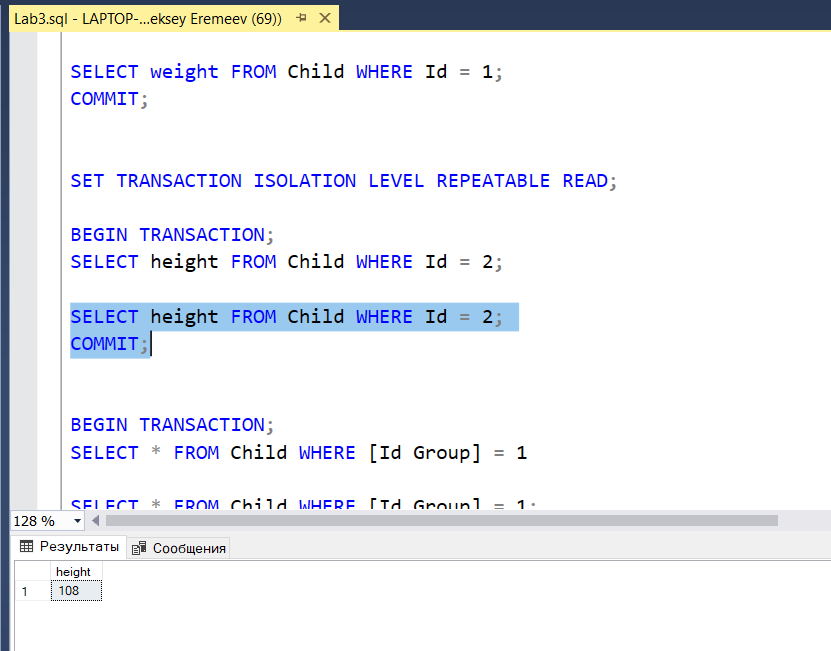
Создаем уровень REPEATABLE READ в обоих сеансах



Создаем в первом сеансе транзакцию первого чтения.

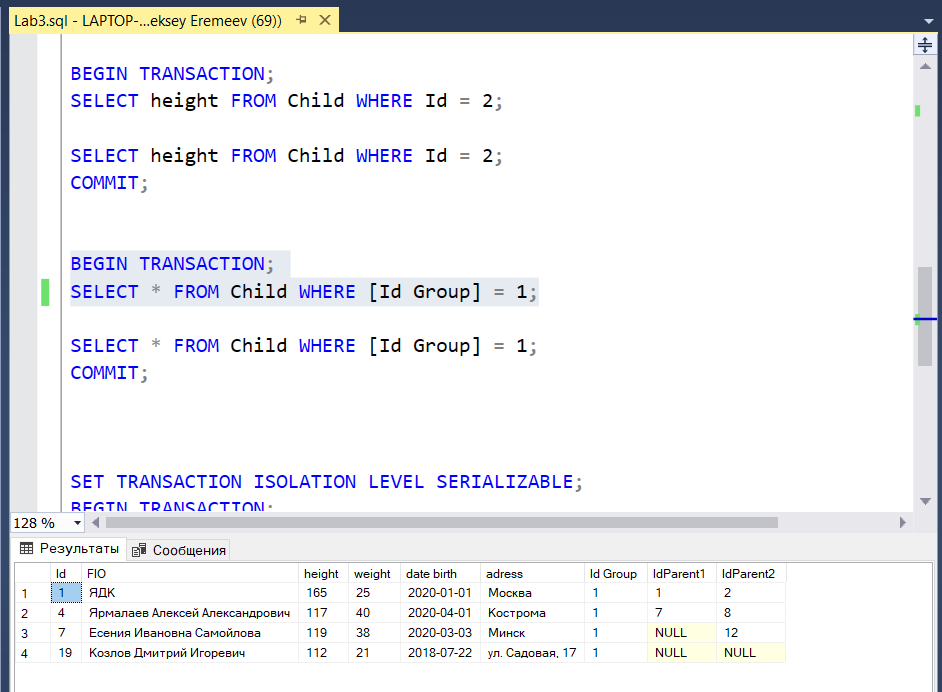


Во втором сеансе создаем транзакцию изменения значения данных и фиксируем их.

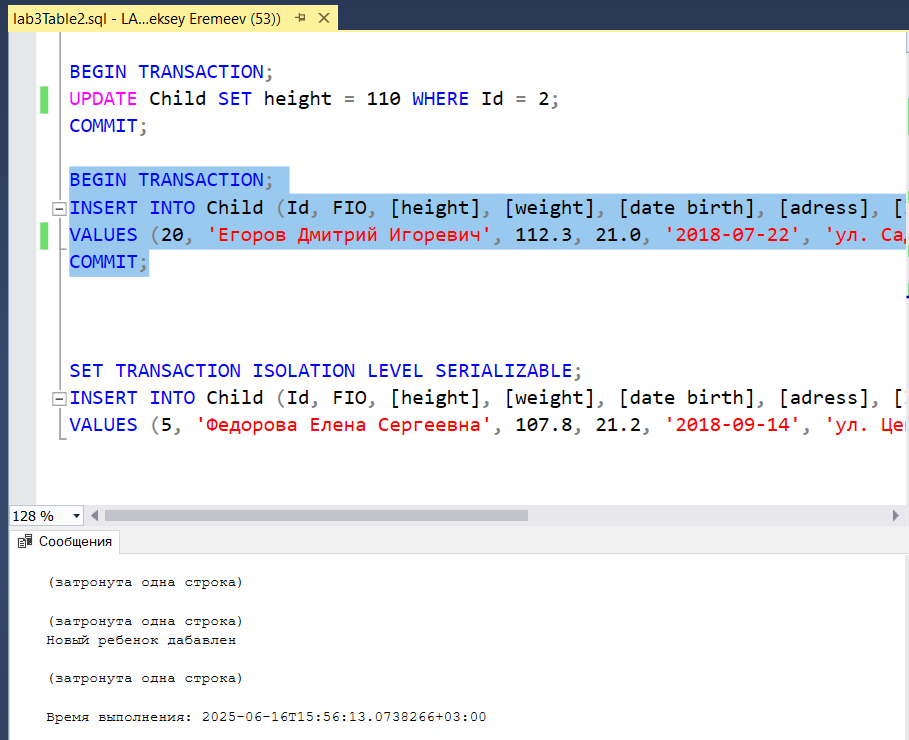


В первом сеансе совершаем второе чтение, и фиксацию транзакции. Что в первом что во втором чтении первого сеанса видны одни и те же данные, что соответствует REPEATABLE READ

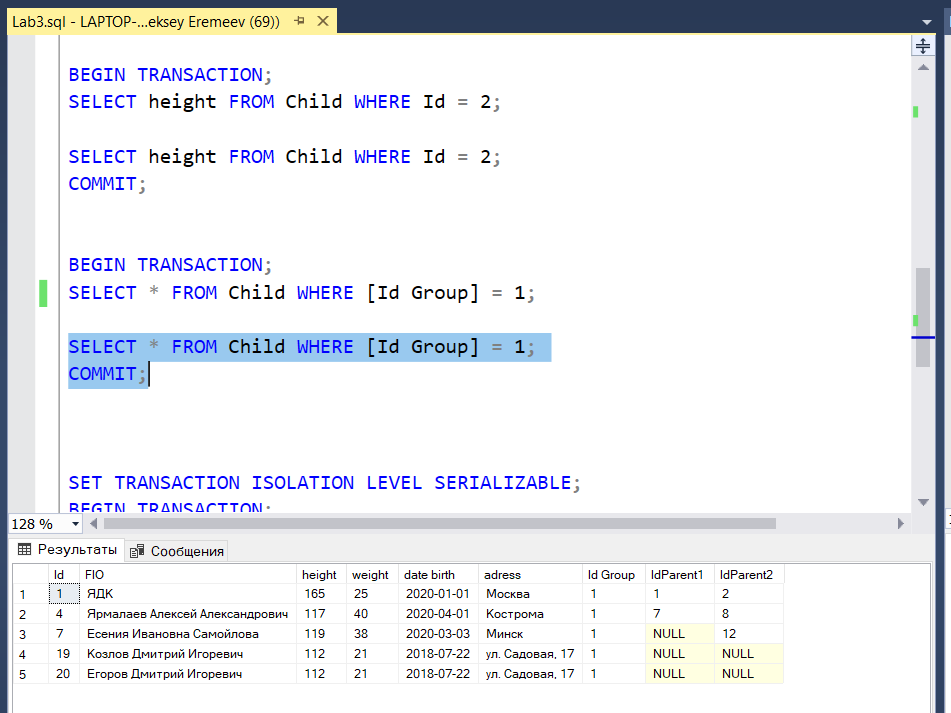
- **Проверка фантомного чтения**



Создаем транзакцию чтения всех данных, без фиксации изменения данной транзакции.



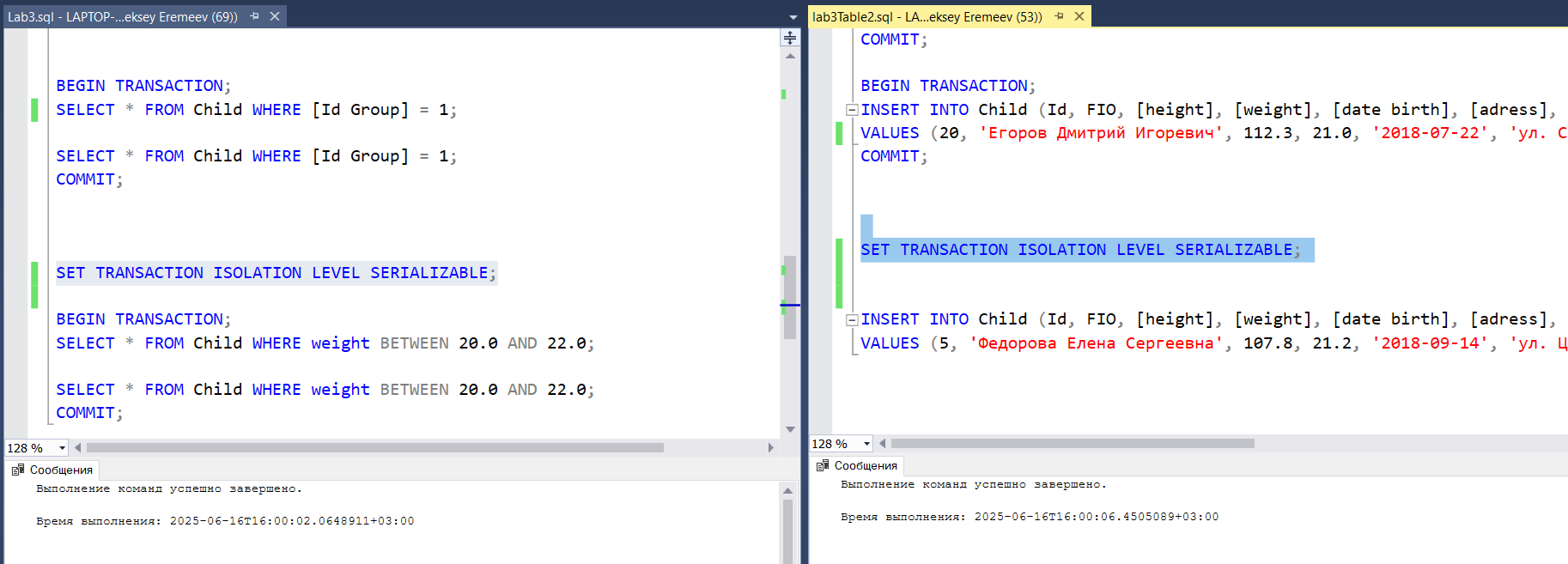
Во втором сеансе создается транзакция с добавлением новых данных и фиксацией данных изменений.



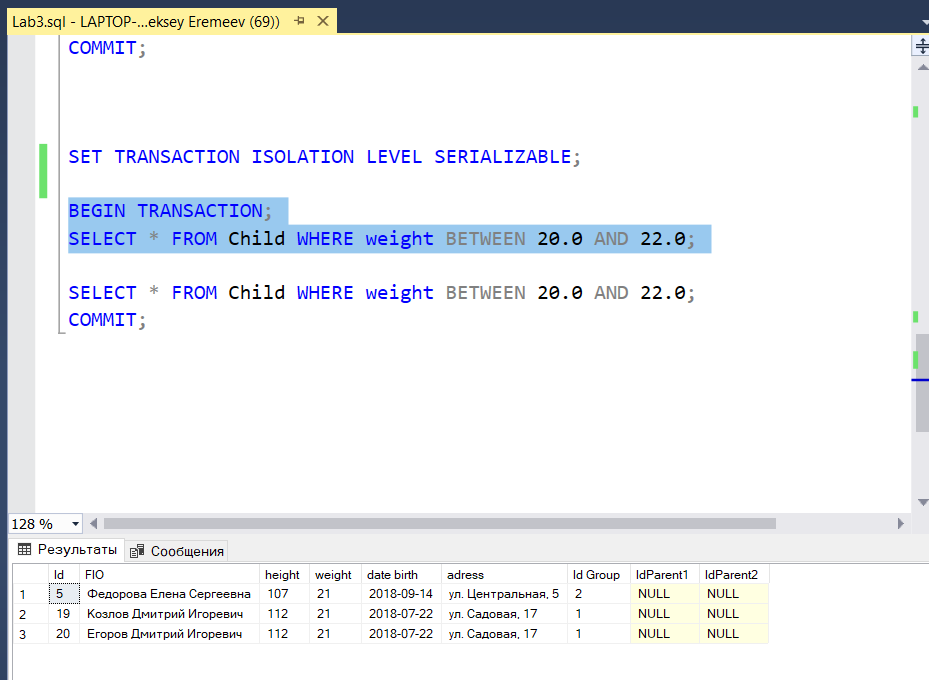
Производим второе чтение первого сеанса и видим добавленные данные из второго сеанса, фиксируем данную транзакцию. Вторая запись чтения – чтение фантомных данных. Что соответсвует REPEATABLE READ

**Уровень изоляции SERIALIZABLE**

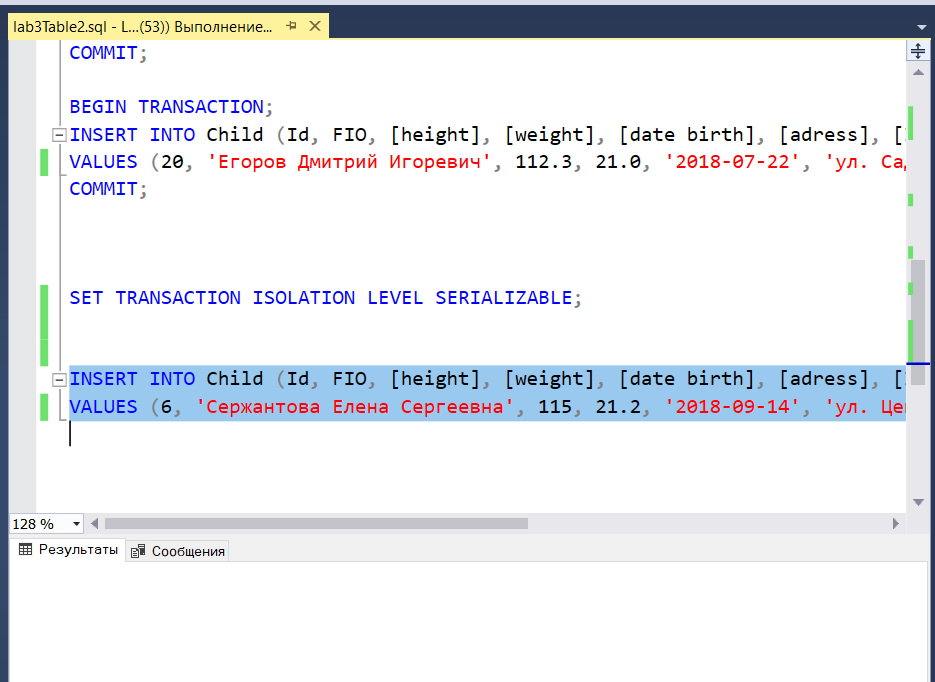
- **Проверка фантомного чтения**



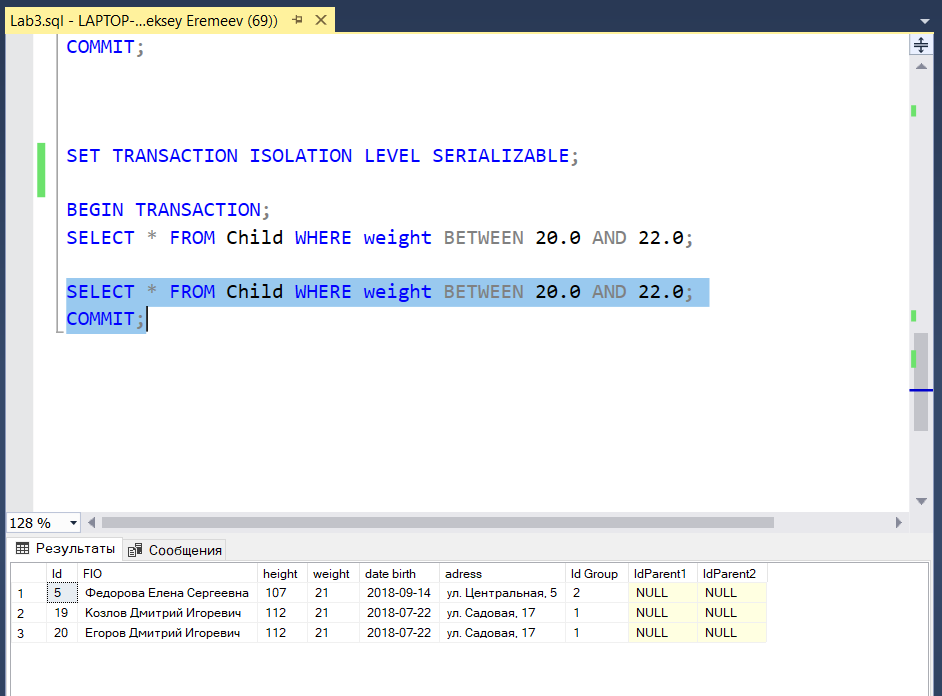
Установка в обоих сеансах уровня изоляции SERIALIZABLE



В первом сеансе создаем транзакцию и совершаем первое чтение, не фиксируя транзакцию.



Во втором сеансе пытаемся добавить новые данные.



В первом сеансе делаем первое чтение, с фиксацией транзакции, где произошла выдача тех же данных. SERIALIZABLE полностью изолировал транзакции, помешав вставку данных, тем самым предотвратив фантомное чтение.