University ID: c7466889

Student Name: Ojaswi Shrestha

Tutorial Tasks Assignment Title: Design/Implementation

**SOLUTIONS:**

**Task 1 – Extended Entity-Relationship Model**

**Holiday Package**

Each holiday package is provided for a certain number of customers. Each holiday package is categorised as being of a specific type e.g. Beach, Winter Sun, Skiing and Adventure, each of which have specific, unique requirements. Each package will fall into one of two journey types, either

* Return flight to a single destination
* Return coach transport with two destinations.

Holidays may have a courier, although some do not.

Where group bookings are made the company needs to record information about the member of the group who is the Group leader; i.e. the arranger of the holiday.

Step 1: Underline all nouns in the text above and list them below:

All the nouns in the question are mentioned below:

* Holiday Package
* Customers
* Type
* Beach
* Winter Sun
* Skiing
* Adventure
* Requirements
* Journey Types
* Destination
* Return Flight
* Return Coach Transport
* Single
* Two
* Couriers
* Group Bookings
* Company
* Information
* Group
* Group Leader
* Arranger

Step 2: Ask yourself - Is your noun

* an attribute of an entity, or
* an occurrence of an attribute or
* an entity, which would have more than ONE occurrences and is it relevant to our system (case study)?

Identifying the nouns below:

* Holiday Package: It is an entity which consists of list of offers regarding the holiday.
* Customers: It is an entity consisting of people details who books the holiday package.
* Type: It is an attribute of Holiday Package entity having specific type of holiday package they are offering.
* Beach: It is an occurrence of the attribute Type in Holiday Package entity.
* Winter Sun: It is an occurrence of the attribute Type in Holiday Package entity.
* Skiing: It is an occurrence of the attribute Type in Holiday Package entity.
* Adventure: It is an occurrence of the attribute Type in Holiday Package entity.
* Requirements: It is an attribute of the Holiday Package entity carrying the unique features for the occurrence of the Type attribute.
* Journey Types: It is an attribute of the Holiday Package entity consist of 2 elements defining the type of transport for the holiday.
* Destination: It is an attribute of Holiday Package entity.
* Return Flight: It is an occurrence of the attribute Journey Types.
* Return Coach Transport: It is an occurrence of the attribute Journey Types.
* Single: It is an occurrence of the Destination attribute.
* Two: It is an occurrence of the Destination attribute.
* Courier: It is an entity which consists of data such as its id, its sender and destination, and its availability in the Holiday package exists or not.
* Bookings: It is an entity consist of the data about the holiday bookings done by people.
* Group: It is an occurrence falling in the attribute of Booking Type of the entity Bookings.
* Company: It is an occurrence falling in the attribute of Booking Type of the entity Bookings.
* Information: It is an occurrence falling in the entity Bookings.
* Group Leader: It is an occurrence falling in the attribute of Travel Organizer of the entity Bookings.
* Arranger: Similar to Group Leader it is also an occurrence falling in the attribute of Travel Organizer of the entity Bookings.

Step 3: For each entity create a table to defining attributes and occurrences.

Holiday\_Package

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Attribute’s Name | Package\_id | Package\_Name | Holiday\_  Type | Requirements | Journey\_Type | Destination | Courier\_id |
| Occurrences | 12 | Snow Fun | Skiing | Age above 10 years | Return Flight | Maldives | 10 |
| Occurrences | 98 | Hot Sun and Cool Sea | Beach | Proper swimwear | Return Coach Transport | Canada | 20 |

Customer\_info

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute’s Name | Customer\_id | Customer\_Name | Customer\_Contact\_No |
| Occurrences | 54 | Smith Johns | 9745126370 |
| Occurrences | 76s | Belle Hermes | 9374105724 |

Courier

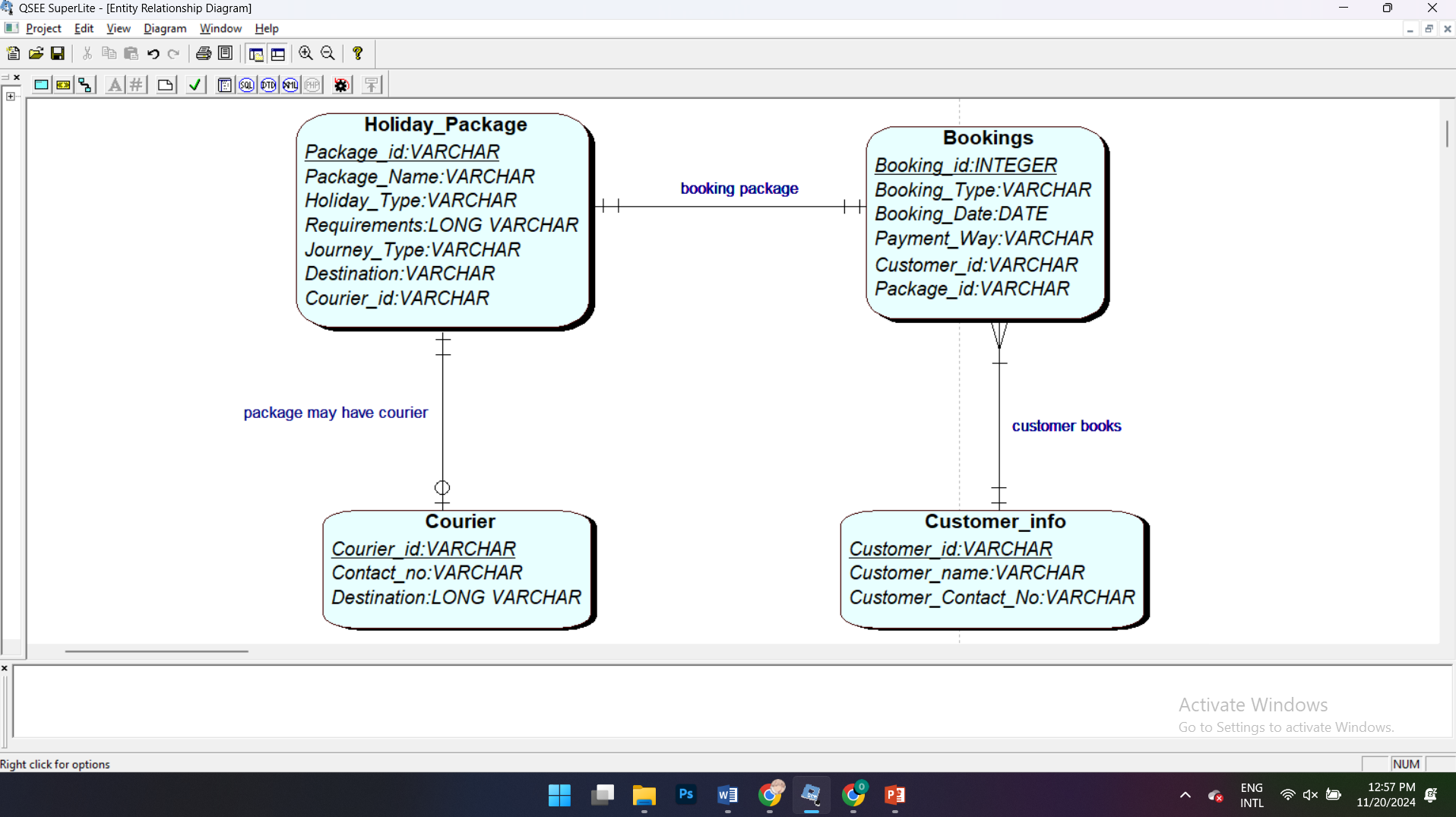
|  |  |  |  |
| --- | --- | --- | --- |
| Attribute’s Name | Courier\_id | Destination | Contact\_no |
| Occurrences | 10 | 456 Elm Street, Suite 3, Los Angeles, CA 90001, USA | 9745126370 |
| Occurrences | 20 | 132, My Street, Big town BG23 4YZss, England | 9374105724 |

Bookings

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Attribute’s Name | Booking\_id | Booking\_Date | Booking\_Type | Customer\_id | Package\_id | Payment\_Way |
| Occurrences | 1 | 23 Dec, 2024 | Company | 54 | 12 | Visa card |
| Occurrences | 2 | 15 June, 2025 | Group | 76 | 98 | Cash |

s

Step 4: Draw your ERD, by defining entities, relationships, relationship names cardinality.



Task 2 (got A, B, C activities) - Logical Design and Normalisation

**A.**

Fill in the missing Entity names.

**Customer**(Custid,

**Order** (Orderid, ….*.Custid*



ii.

**Order** (Orderid,

**Item** (Itemid,…*Order\_id*

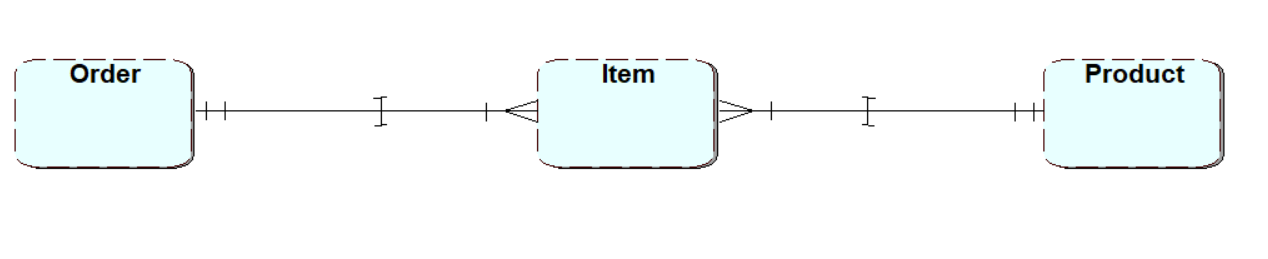
**

iii.

**Order** (Orderid,

**Product** (Product\_id

**Item** (Itemid,…***Order\_id, Product\_ids***



1. In this task you are asked to produce an ERD using the given relations and the keys (logical design in reverse).

**Department** (Dept\_id, …)

**Employee** (Emp\_id,….., *Dept\_id)*

**Job** (Job\_id,….)

**Shift (***Job\_id, Emp\_id,….)*

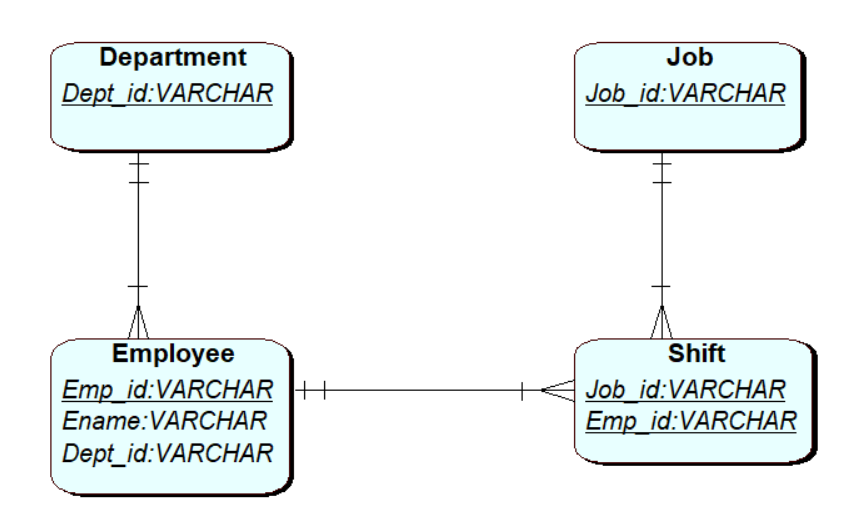


Fig: ERD of Employee table

**B.**

Derive this model fully by defining ***foreign keys*** for the relevant tables.

Below is the ERD’s description:

*Every project requires a number of employees. It is usual for consultants to work on more than one project at a time. Each project is broken down into individual tasks which are allocated to specific consultants. Some tasks are common to any project. Consultants meet regularly with clients by making appointments with them.*

Note: When you are completing the task to derive keys from your ERD, no retrospective changes should happen to the ERD design.



Fig: ERD of the Project table

**C. HOUSE LETS – Normalisation**

The Agency undertakes regular inspections of the properties. Staff are allocated a company pool car for the day. A member of staff may inspect many properties in one day, but a property is inspected only once in any one day.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Page No: 1 | | Acme Letting Agency  Property Inspection Report | | | | | Date: | 12/10/96 | |
| Property No: p103 | | | | Property Address: | | | | | |
| Inspection Date | Inspection Time | | Comments | | Staff Number | Staff Name | | | Car Reg. |
|  |  | |  | |  |  | | |  |
|  |  | |  | |  |  | | |  |

Your task is to identify which Normal Form has been incorrectly normalised.

|  |  |  |  |
| --- | --- | --- | --- |
| **UNF** | **1NF** | **2NF** | **3NF** |
| Property No. | Property No. | Property No. | Property No. |
| Property Addr. | Property Addr. | Property Addr. | Property Addr. |
| Idate\* |  |  |  |
| Itime \* | *Property No.* | *Property No.* | *Property No.* |
| Comment\* | Idate | Idate | Idate |
| Staff\_no\* | Itime | Itime | Itime |
| Sname\* | Comment | Comment | Comment |
| Car\_reg\* | Staff\_no | *Staff\_no* | *Staff\_no* |
|  | Sname |  |  |
|  | Car\_reg |  |  |
|  |  | Staff\_no | Staff\_no |
|  |  | Sname | Sname |
|  |  | Car\_reg | Car\_reg |

Solution:

There is problem with 2NF and 3NF because:

In 2NF, partial dependency is not resolved (Property Address is still dependent on Property No)

In 3NF, Transitive dependencies is not resolved (Staff Name still depends on Staff

Number)

Complete normalization format

* Second normal form (2NF):
  + - Property table

|  |  |
| --- | --- |
| Property no | Property Addr. |

* + - Inspection table

|  |  |  |  |
| --- | --- | --- | --- |
| Property no | Inspection Date | Inspection Time | Comment |

* + - Staff allocation table

|  |  |  |  |
| --- | --- | --- | --- |
| Inspection date | Inspection time | Staff no | Car Reg |

* + - * + Partial dependence on Property no is removed by creating separate table for property information.
        + All non-key attributes in each table depends on primary key of that table.
* Third normal form(3NF):
  + - Property table

|  |  |
| --- | --- |
| Property no | Property Addr. |

* + - Inspection table

|  |  |  |  |
| --- | --- | --- | --- |
| Property no | Inspection Date | Inspection Time | Comment |

* + - Staff allocation table

|  |  |  |  |
| --- | --- | --- | --- |
| Inspection date | Inspection time | Staff no | Car Reg |

* + - Staff table

|  |  |
| --- | --- |
| Staff no | Staff Name |

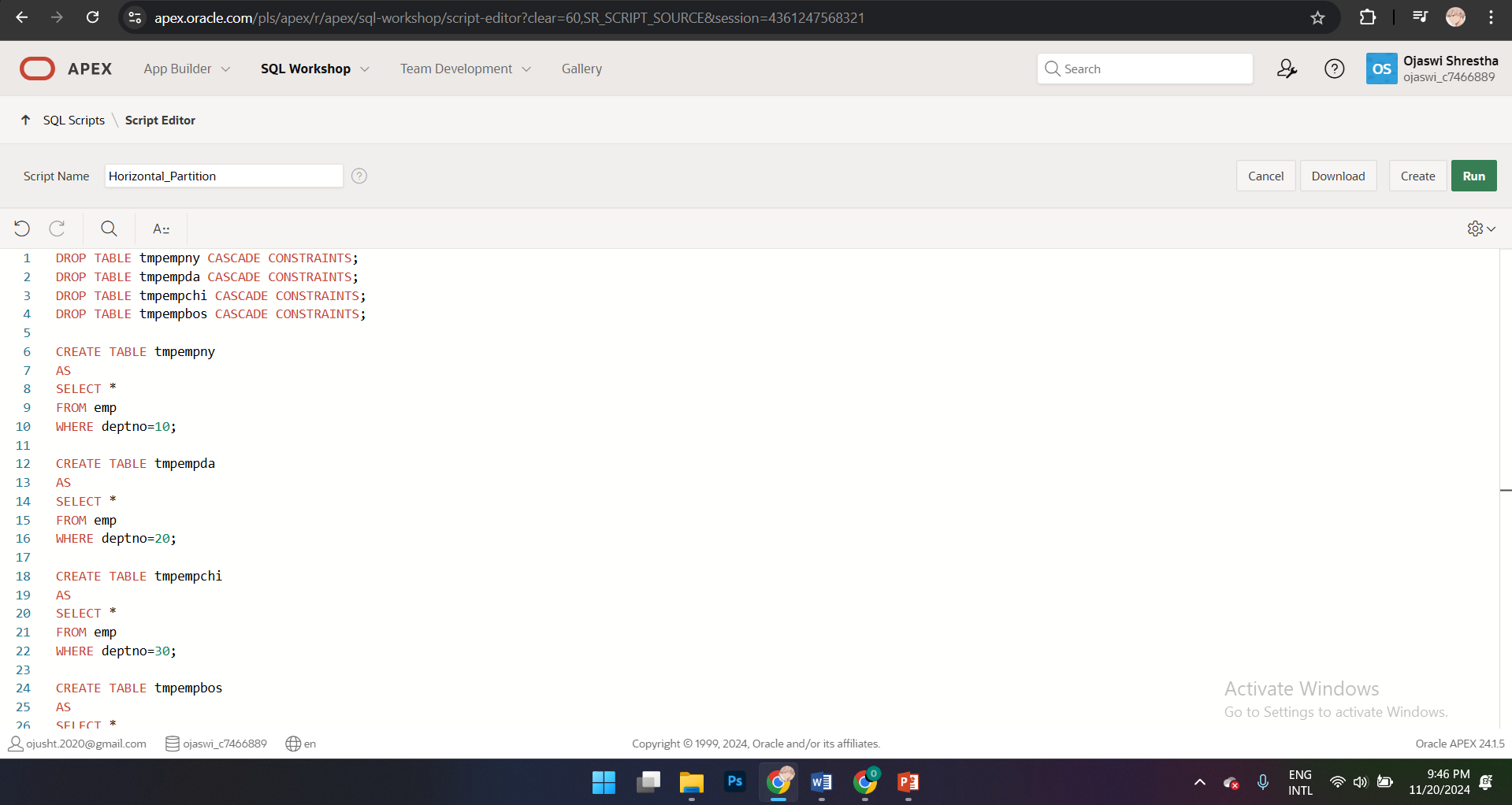
* + - * + Transition dependency on staff no is removed by creating new table which holds information off staff information.
        + Non key attribute on each table depends only on primary key

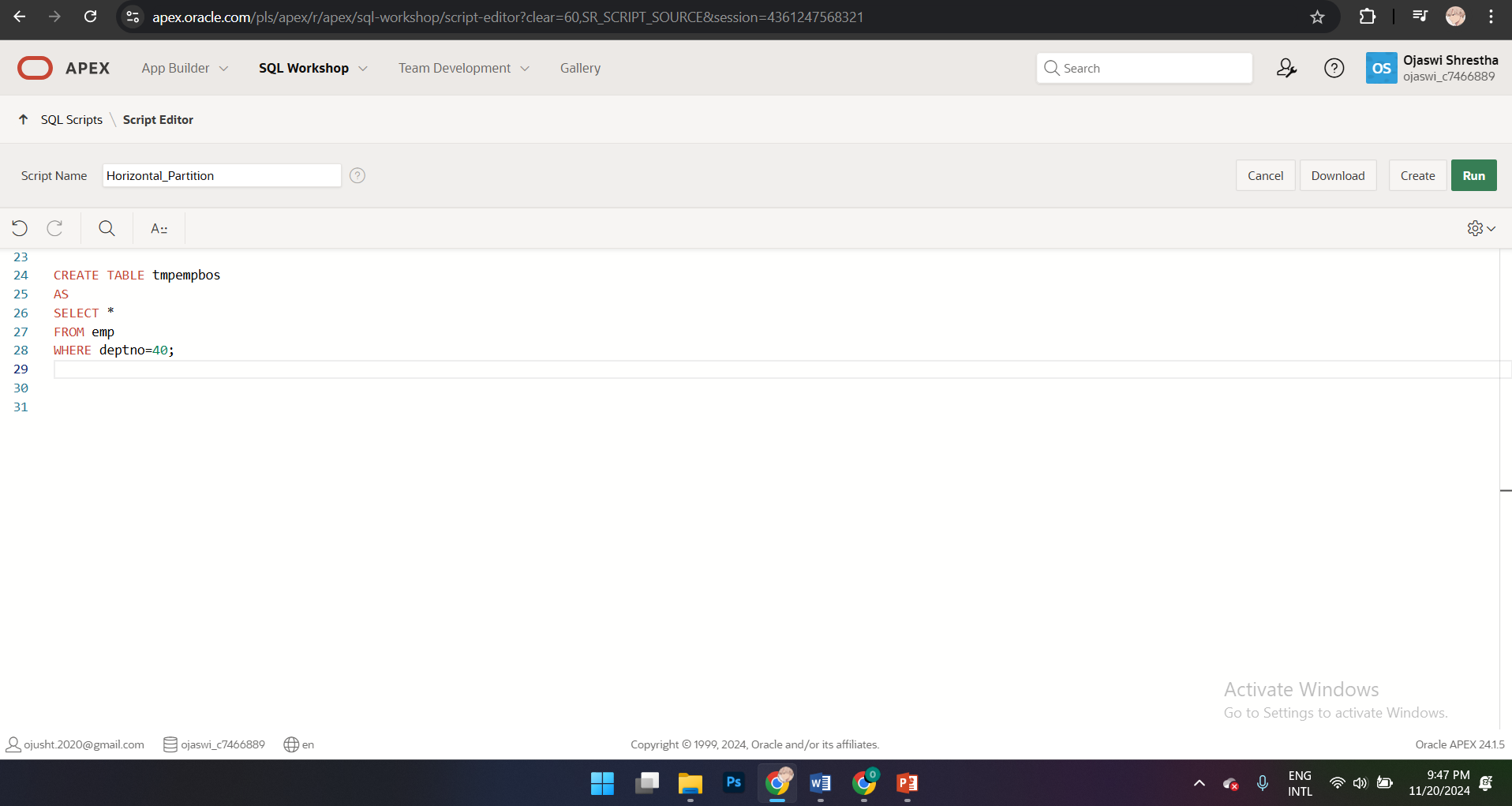
**Task 3** **- Physical Design**

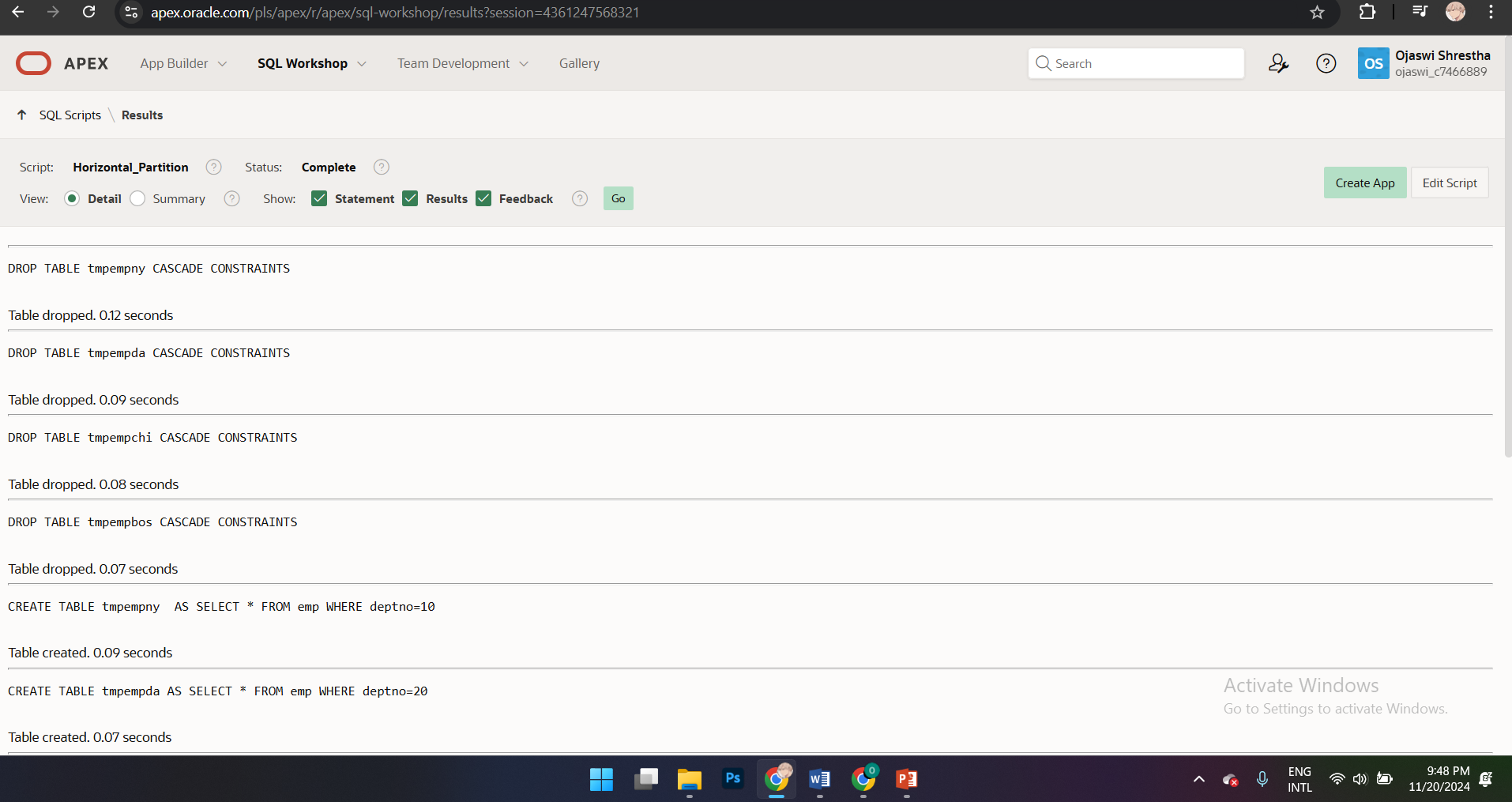
**Complete the Physical Desing SQL tutorial – based on scott tables.**

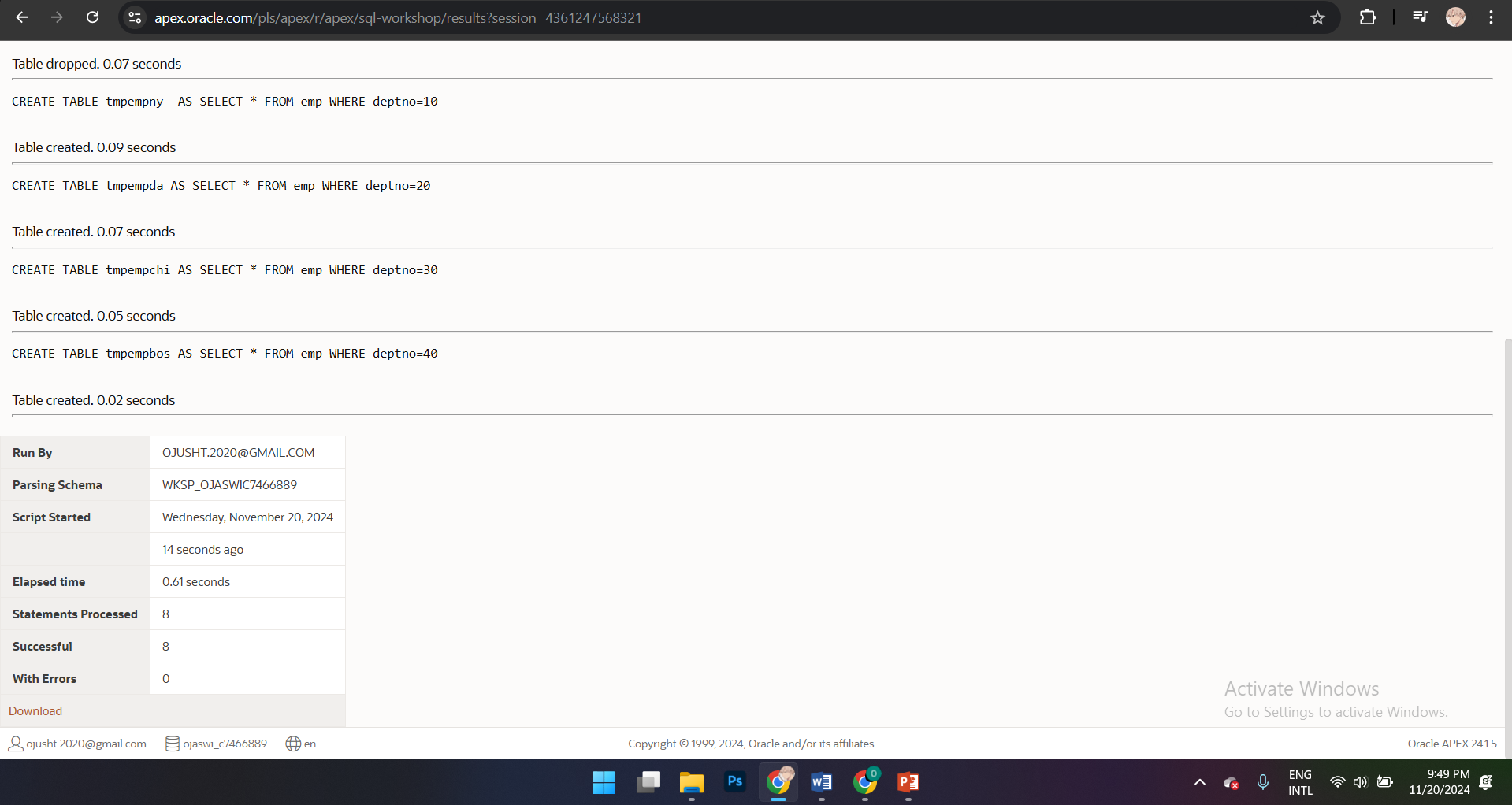


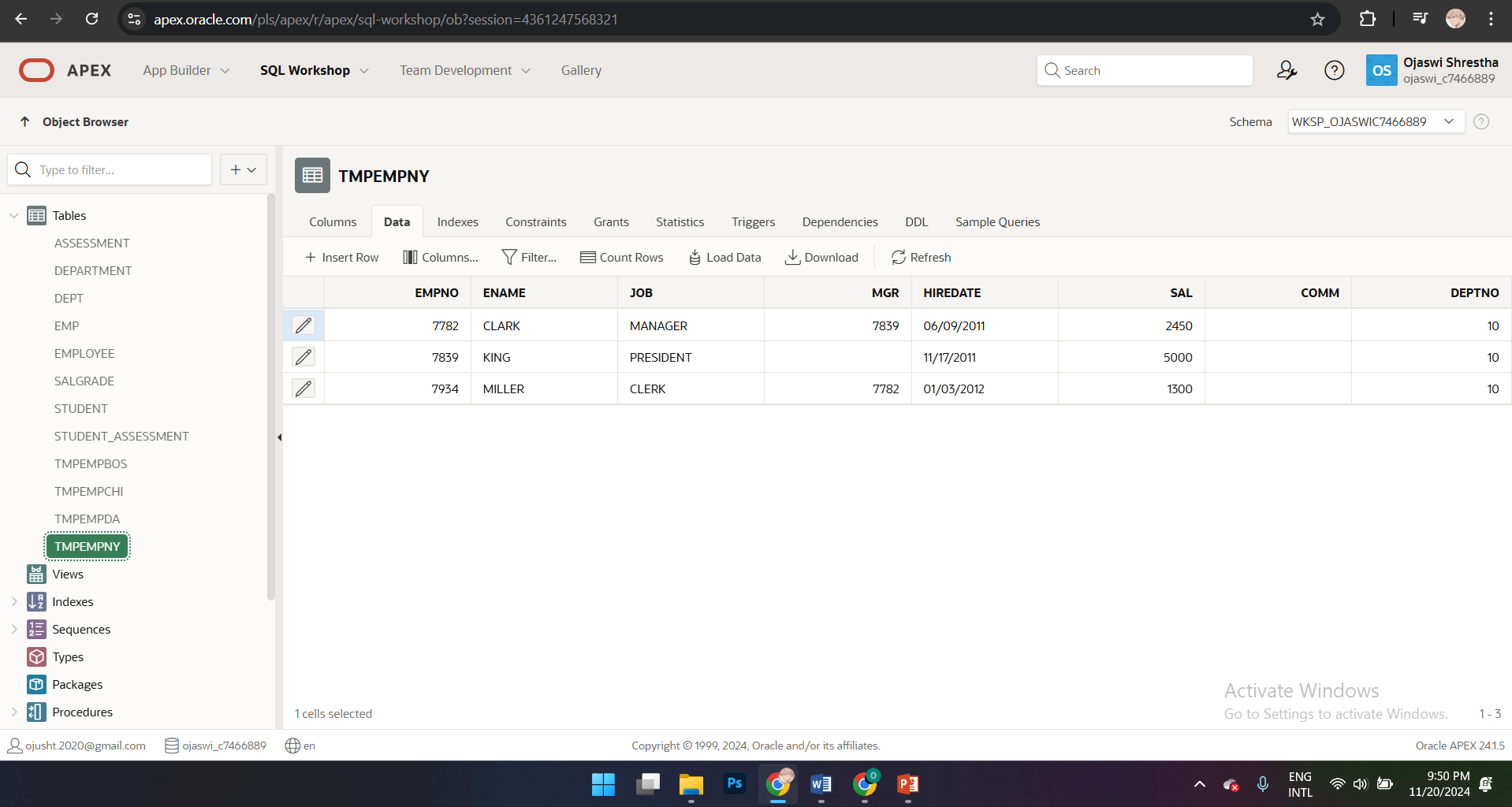
1. Notice departments are in different cities. (horizontal partitioning)
   1. Write SQL to create 4 ‘emp’ tables, one for each city eg tmpempny, tmpempda. Check you haven’t lost any data. Create this as a script (drop tables before creating to make it re-runnable).

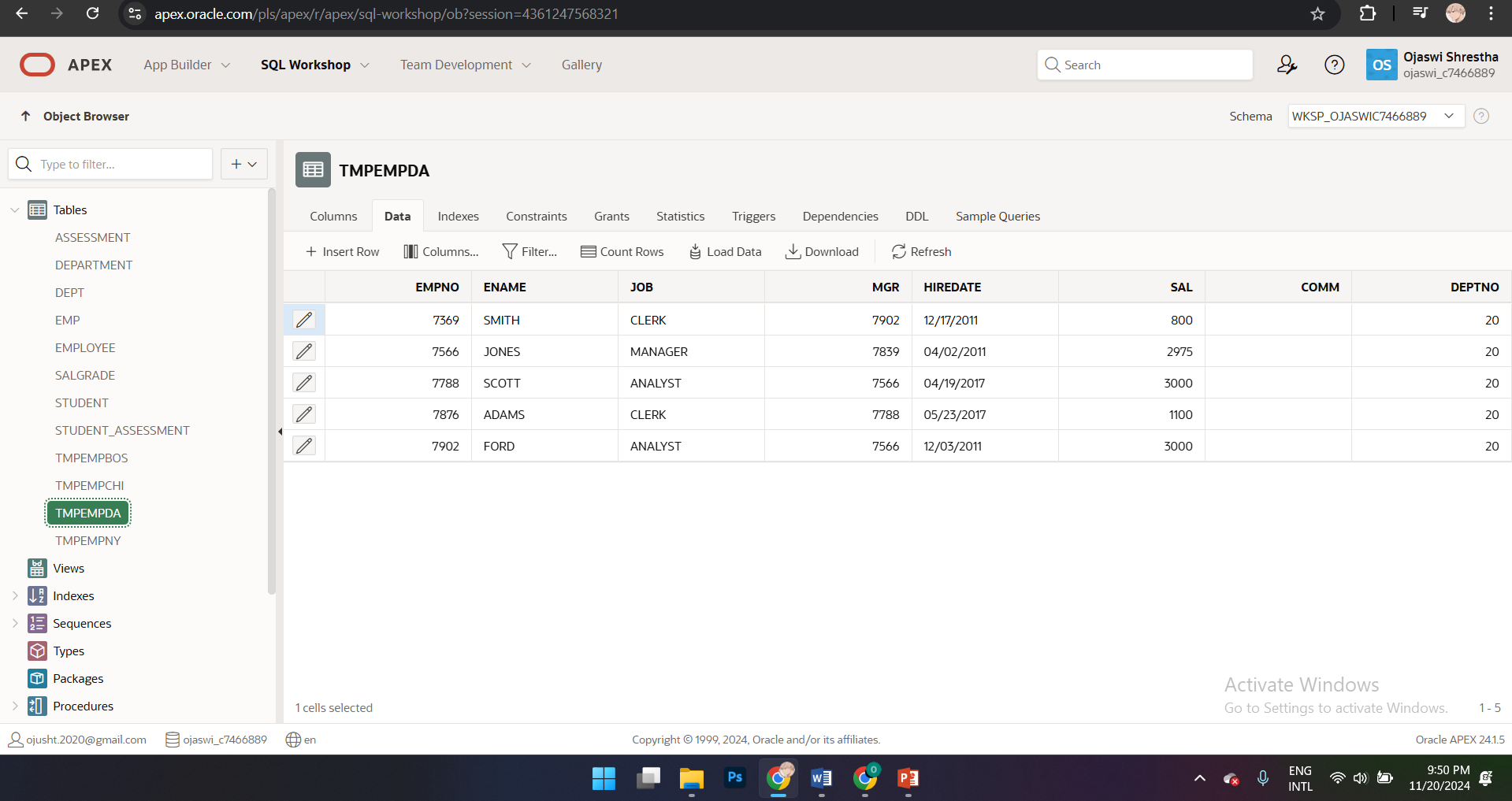


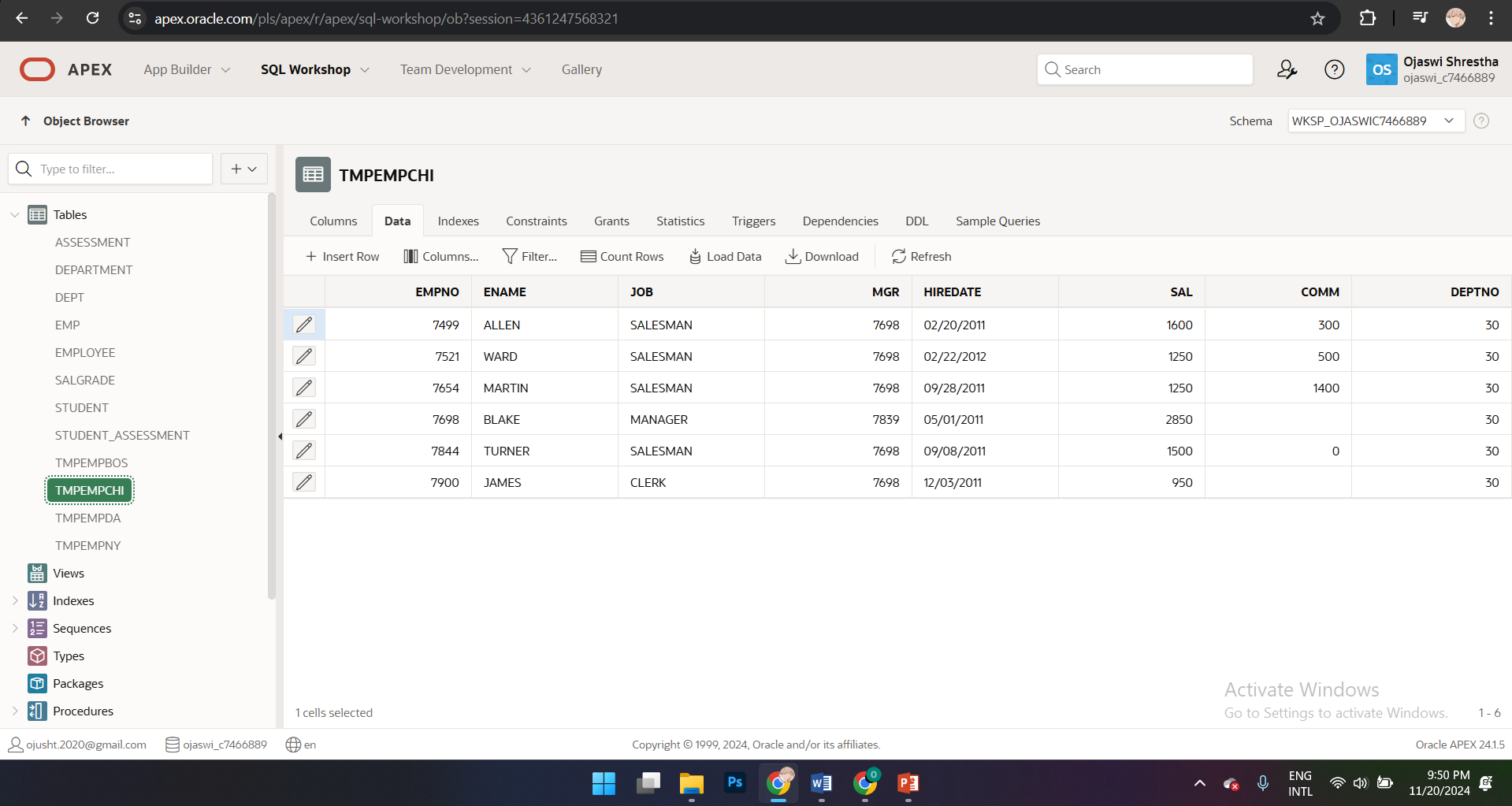


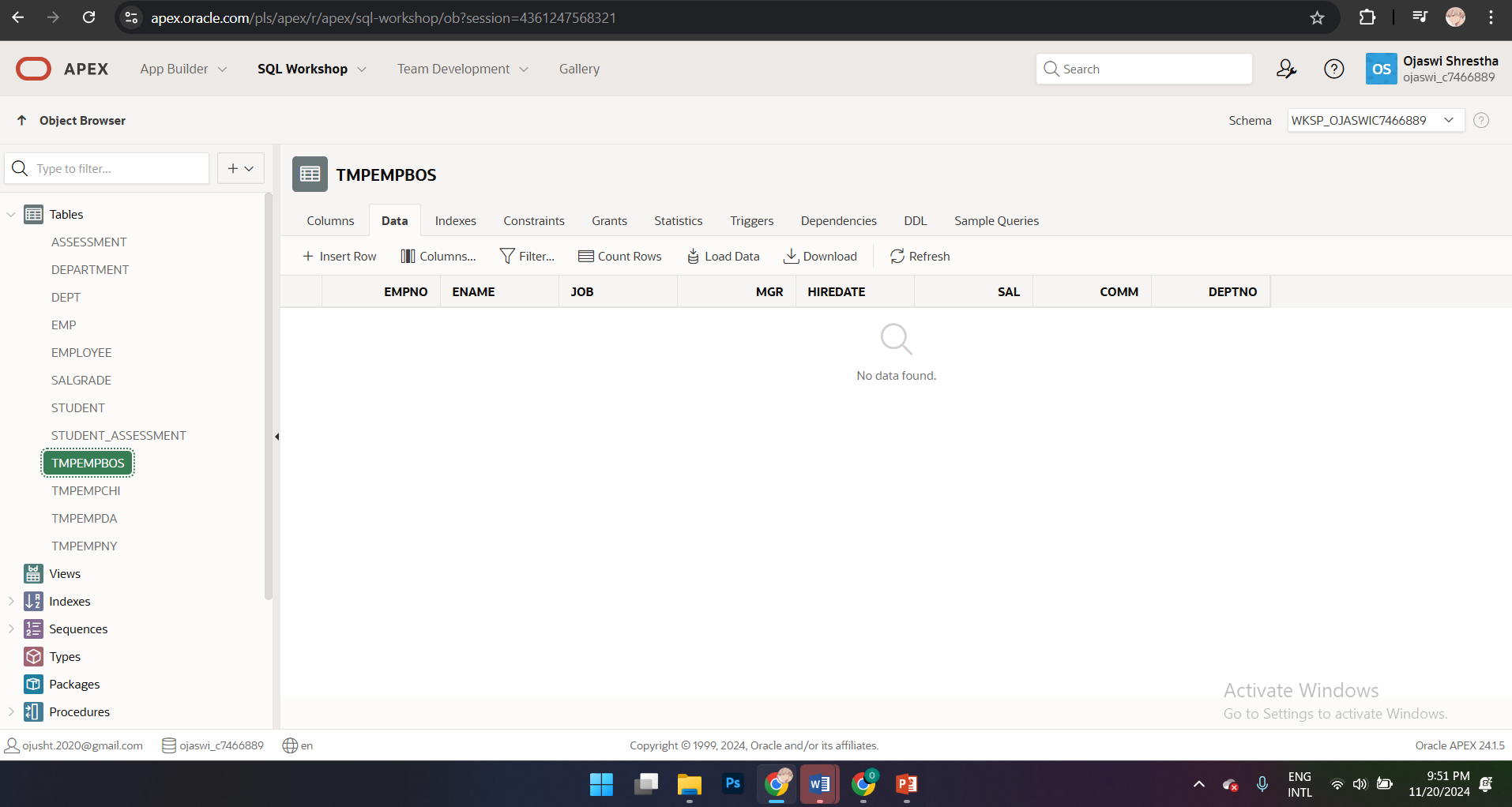






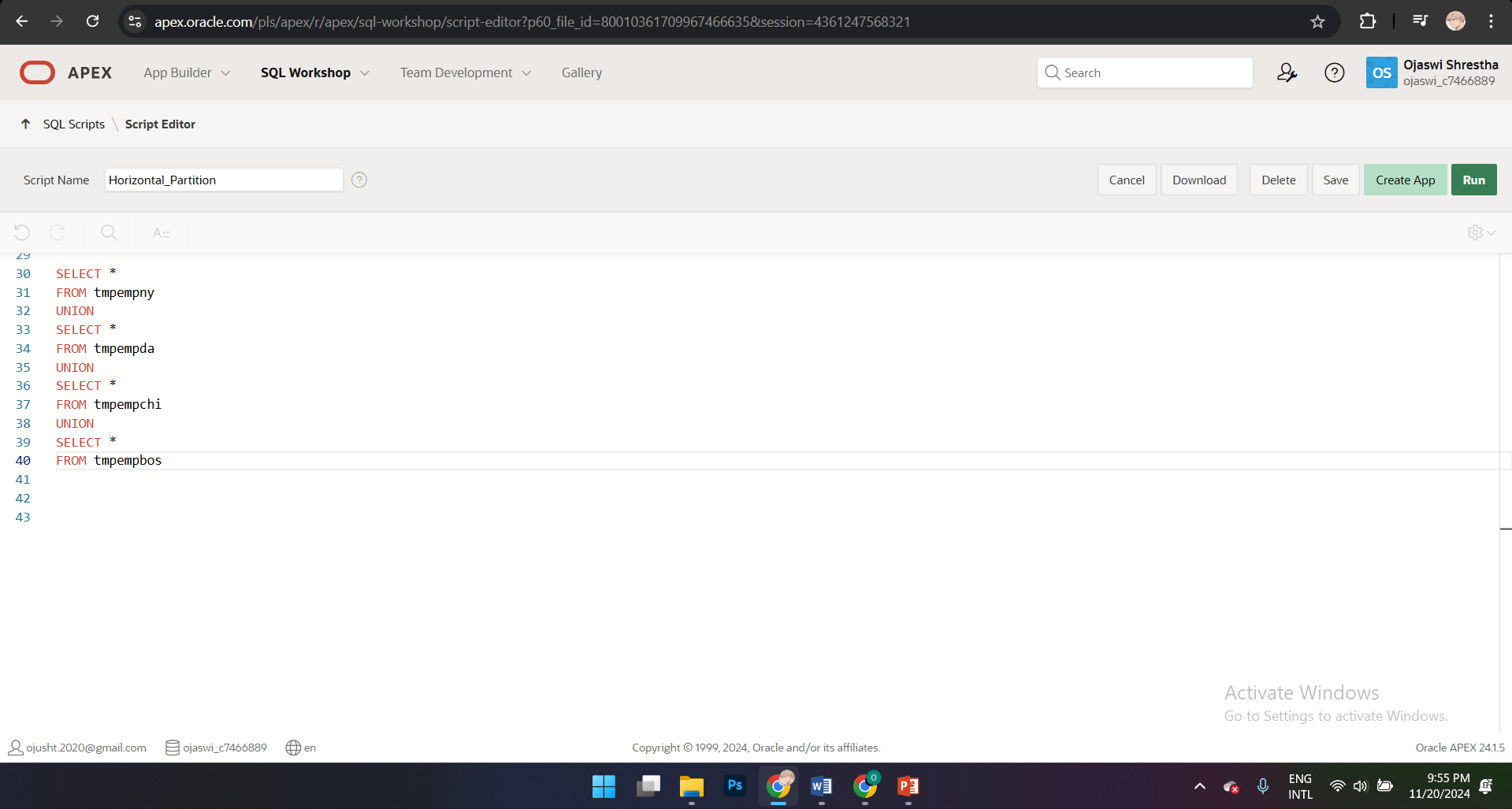


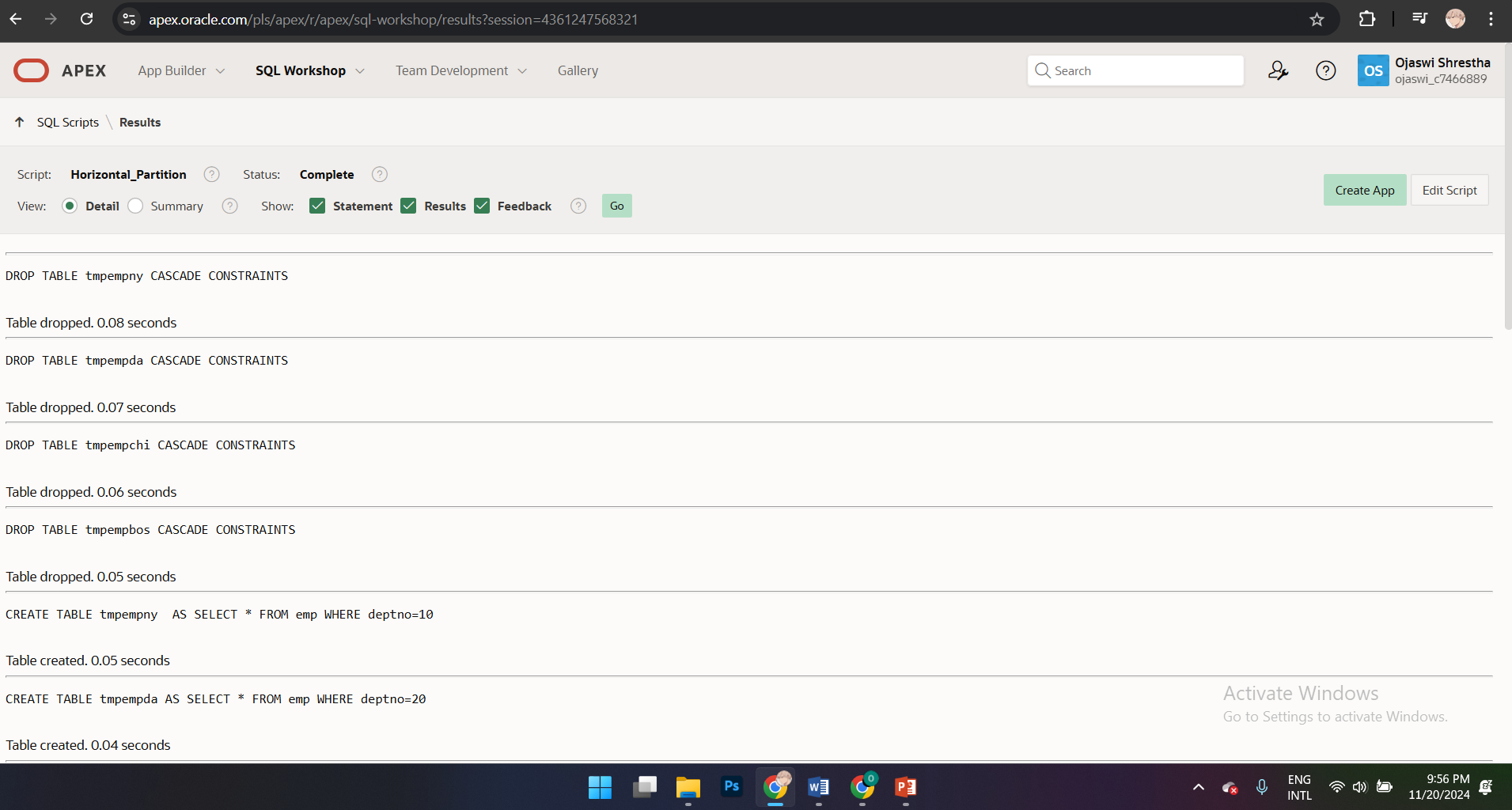


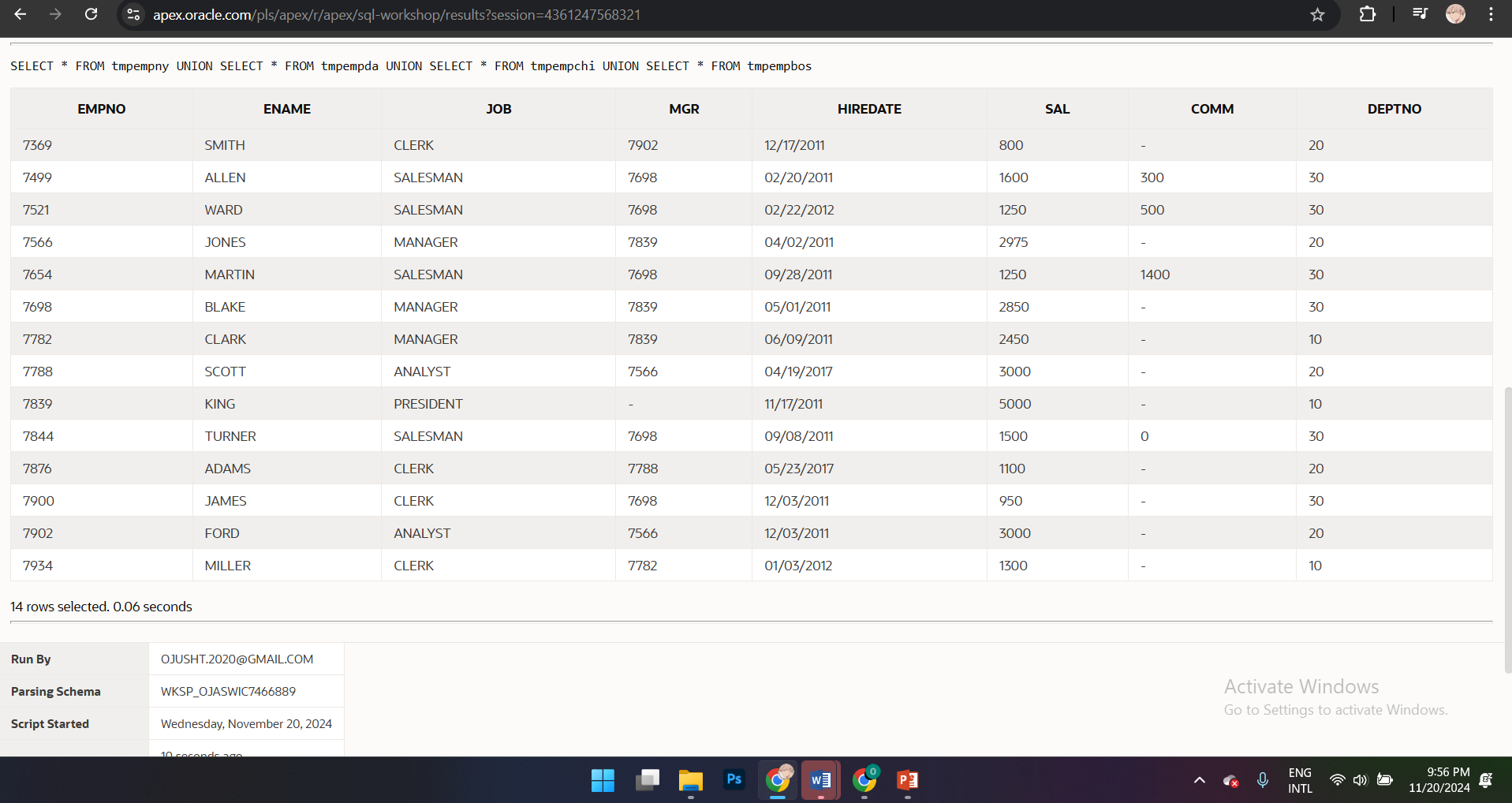


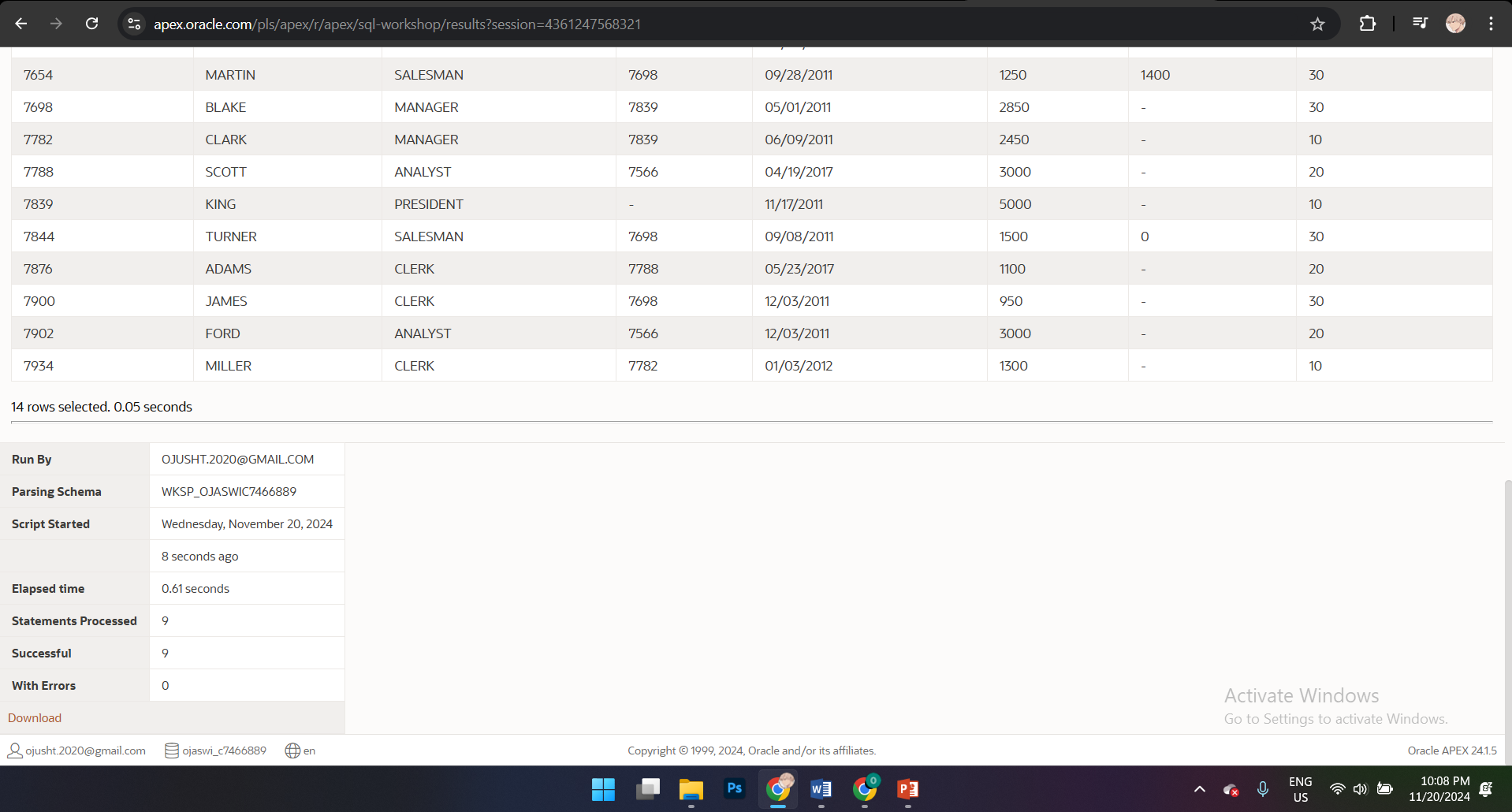
* 1. 1a) Now write SQL to select all employees from the Company.

Hint: use the UNION command. UNION compatibility is important



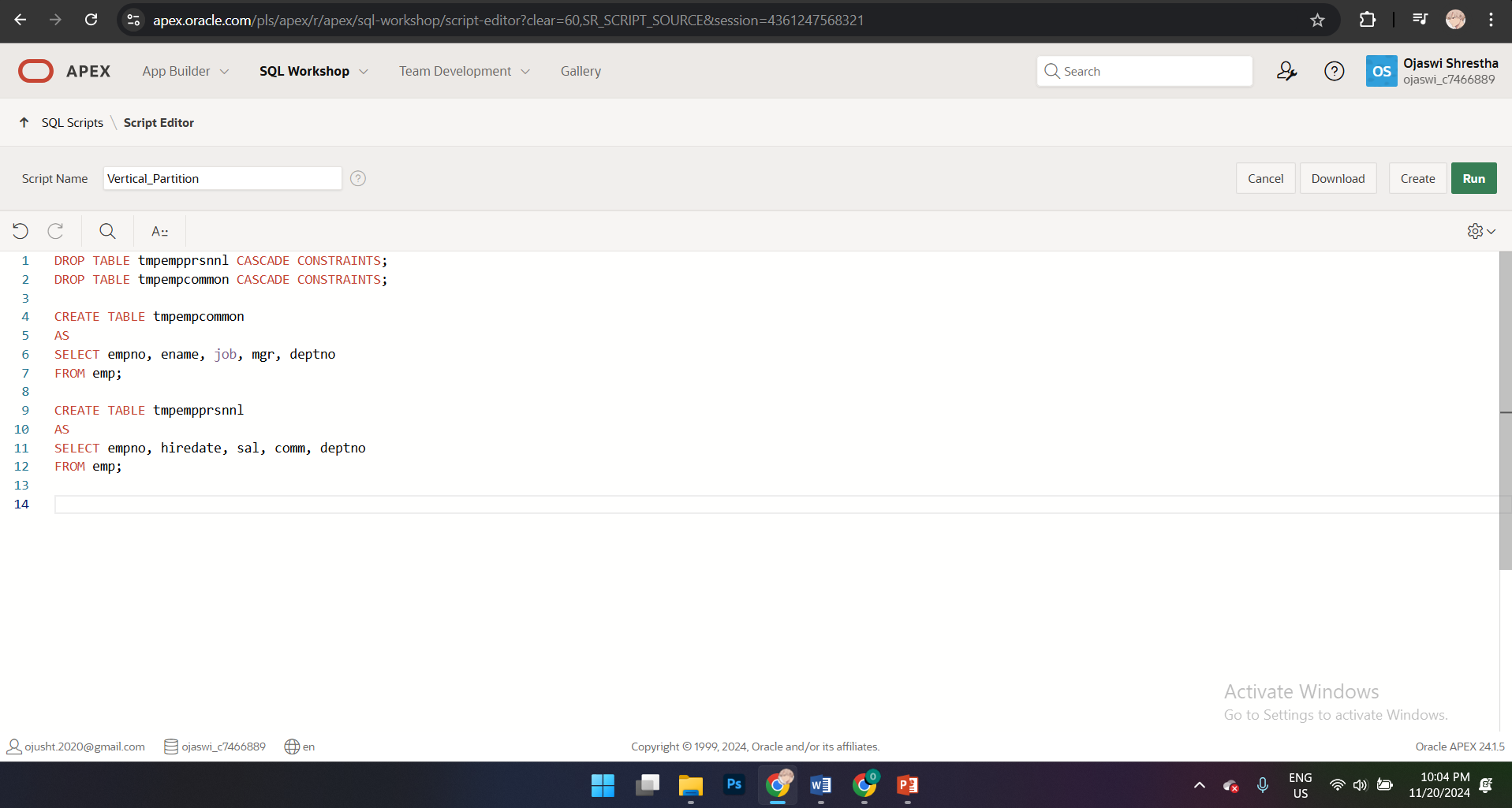


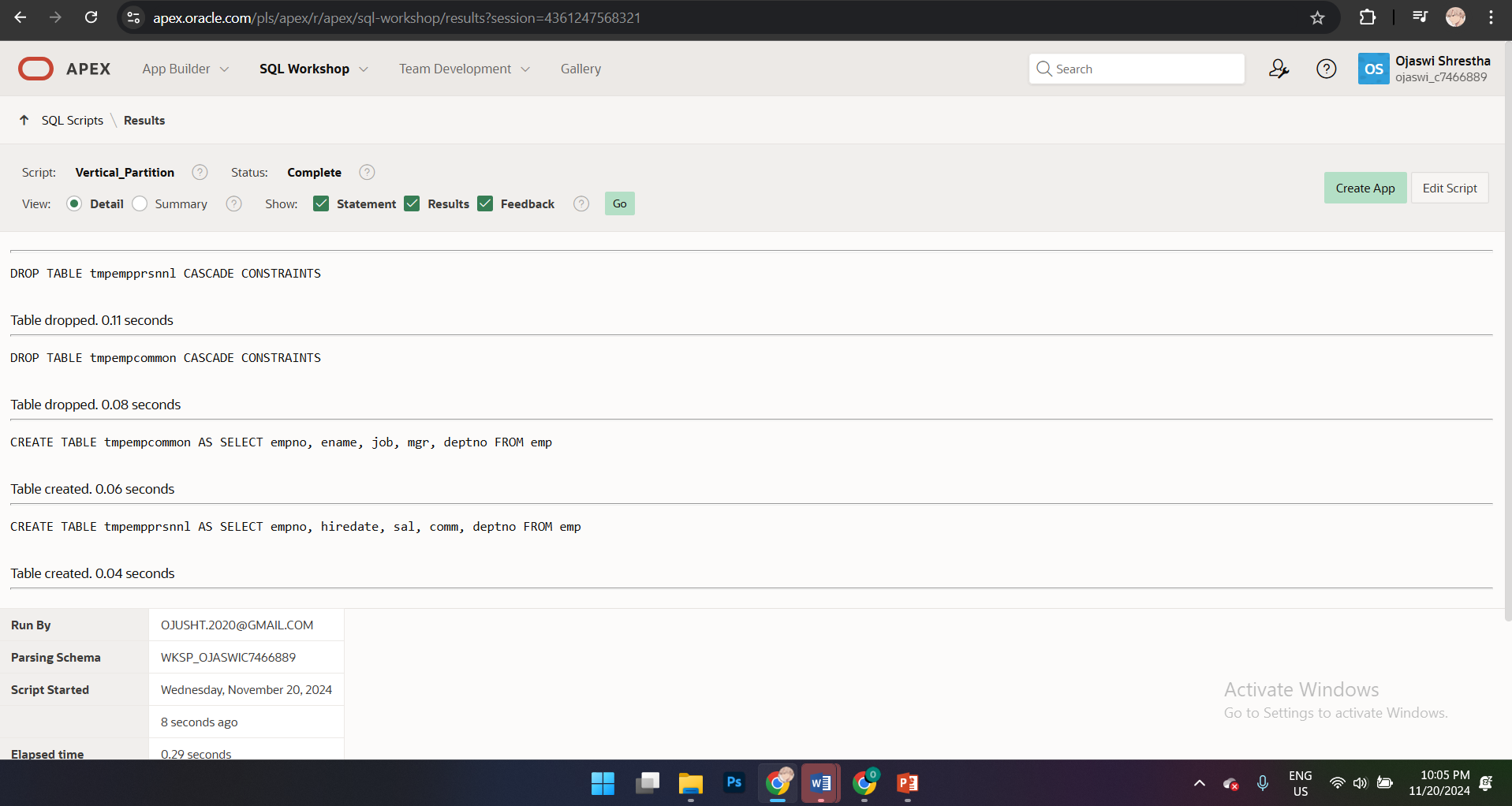


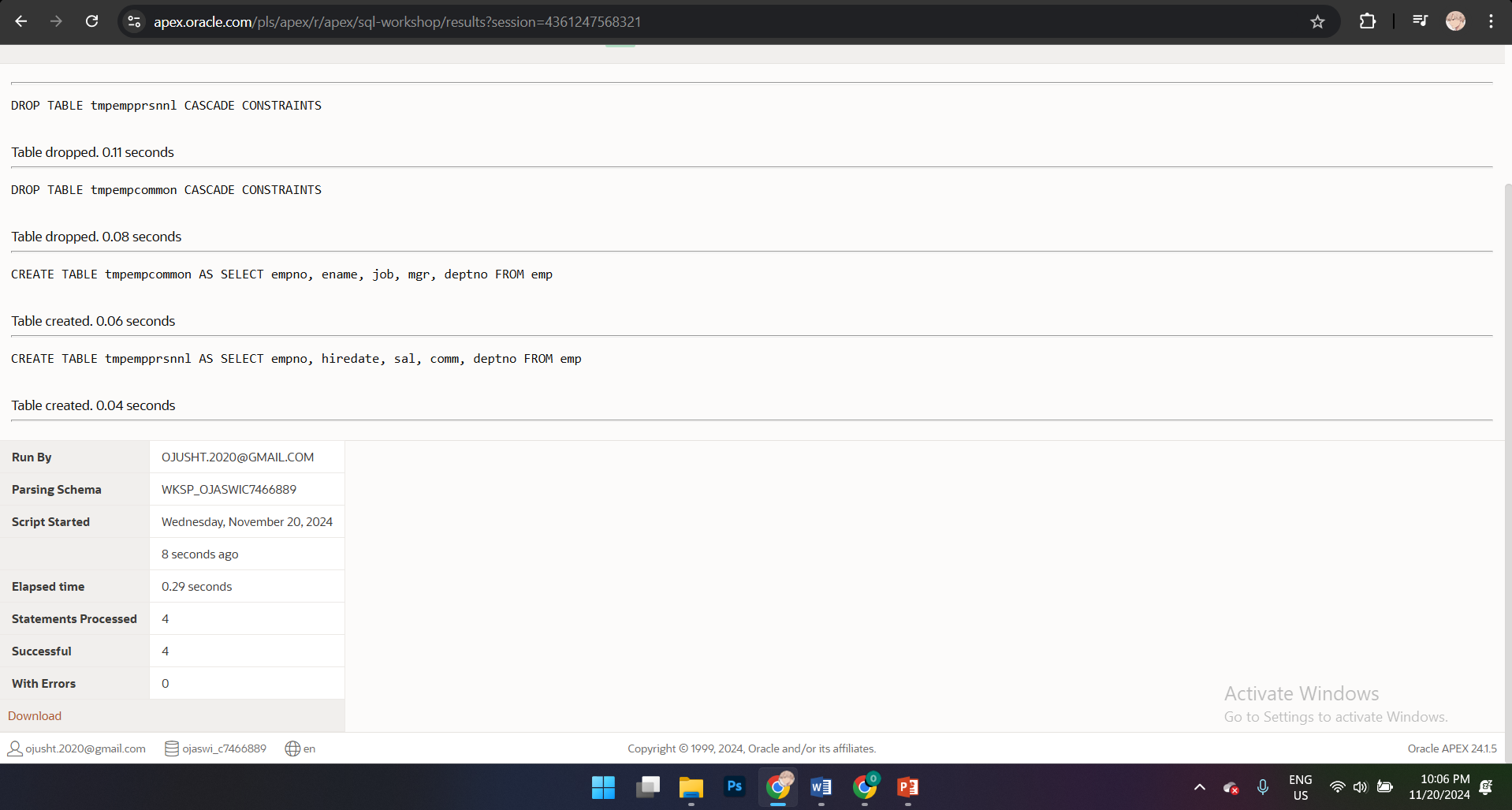


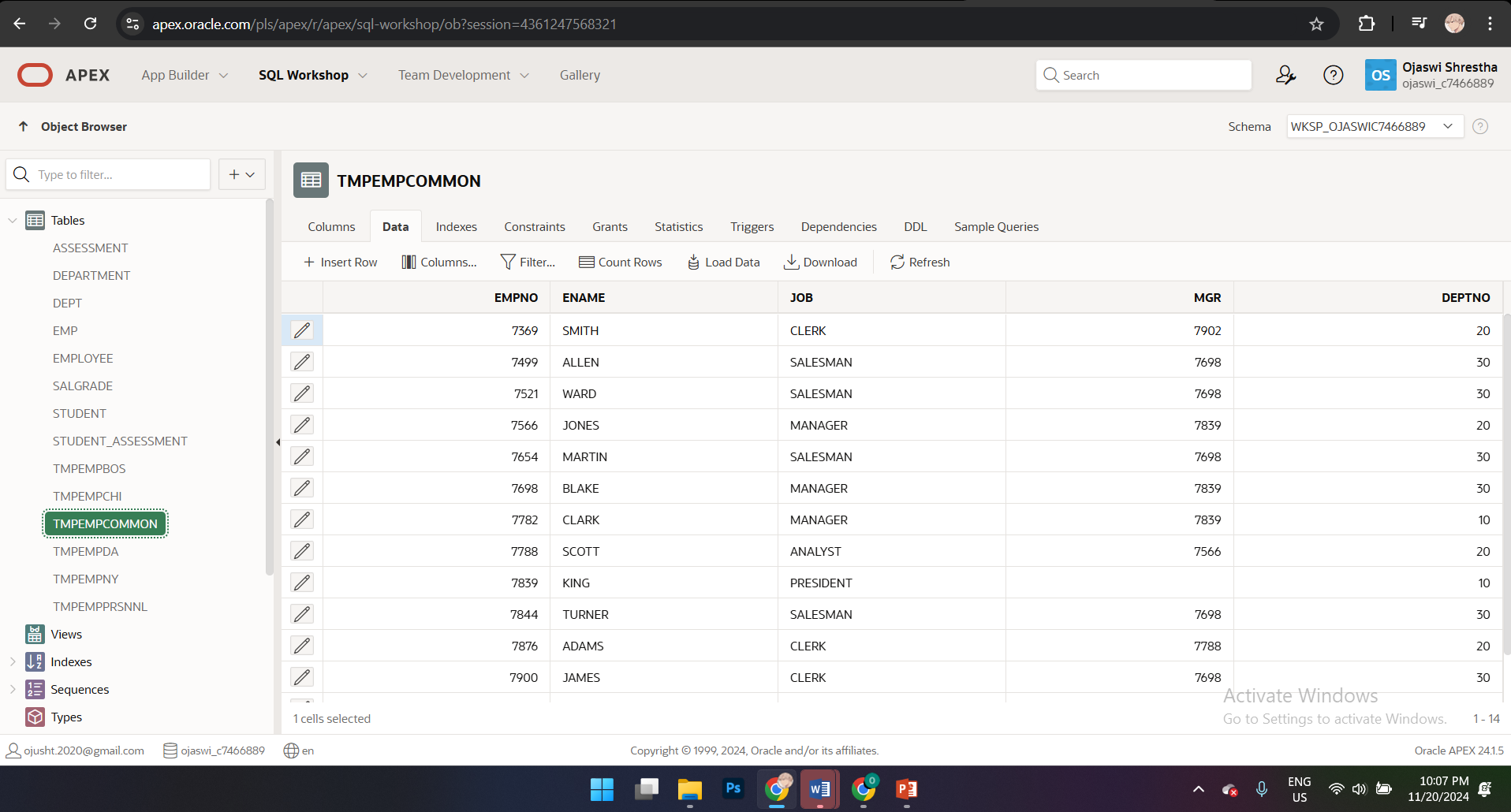
1. Assume we need to hold sensitive employee data on one table ‘tmpempprsnnl’ and non sensitive data on another table tmpempcommon.

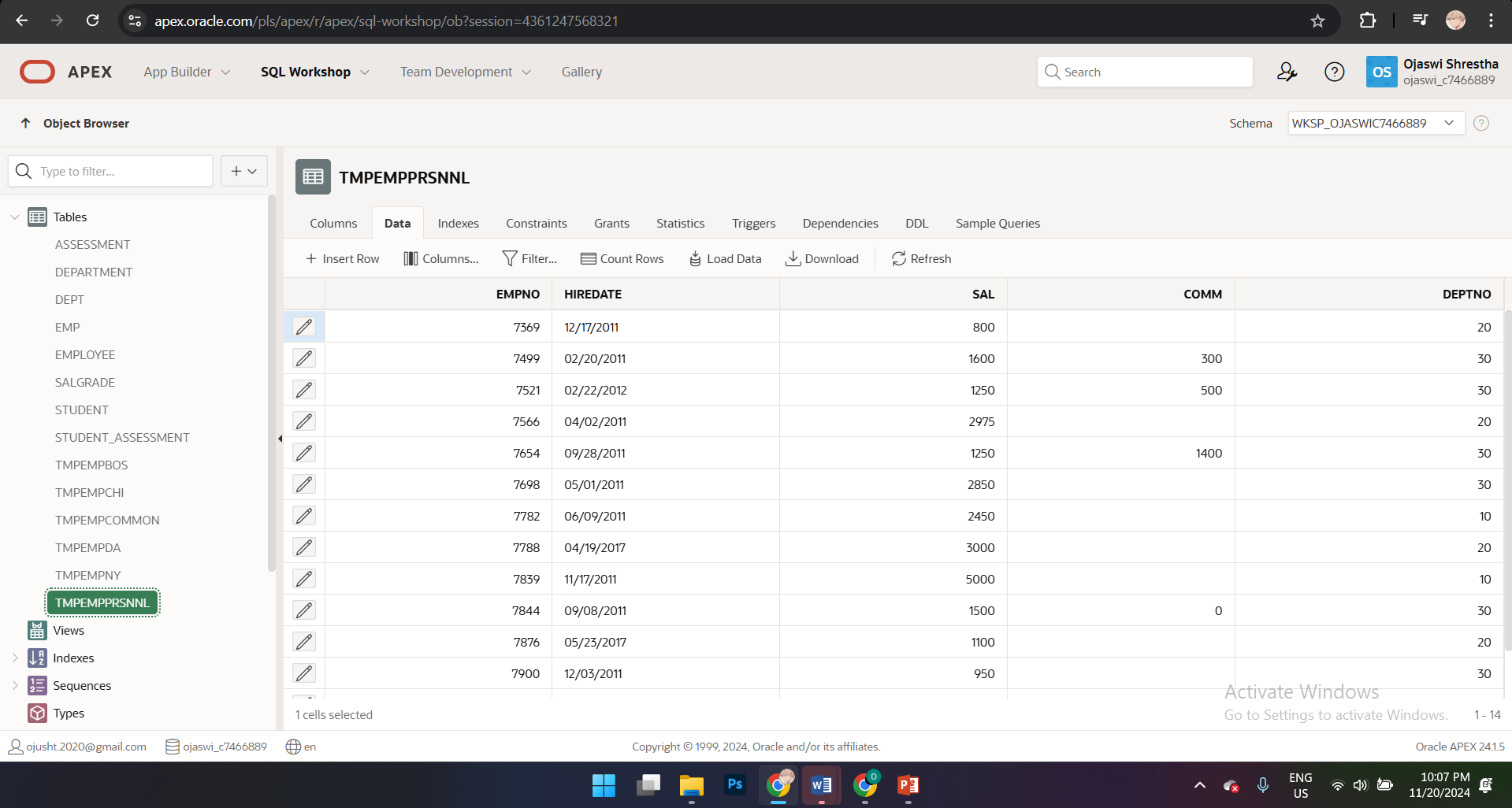
Write the SQL to do this. Check the data is correct. Create it as a re-runnable script. (vertical partitioning).



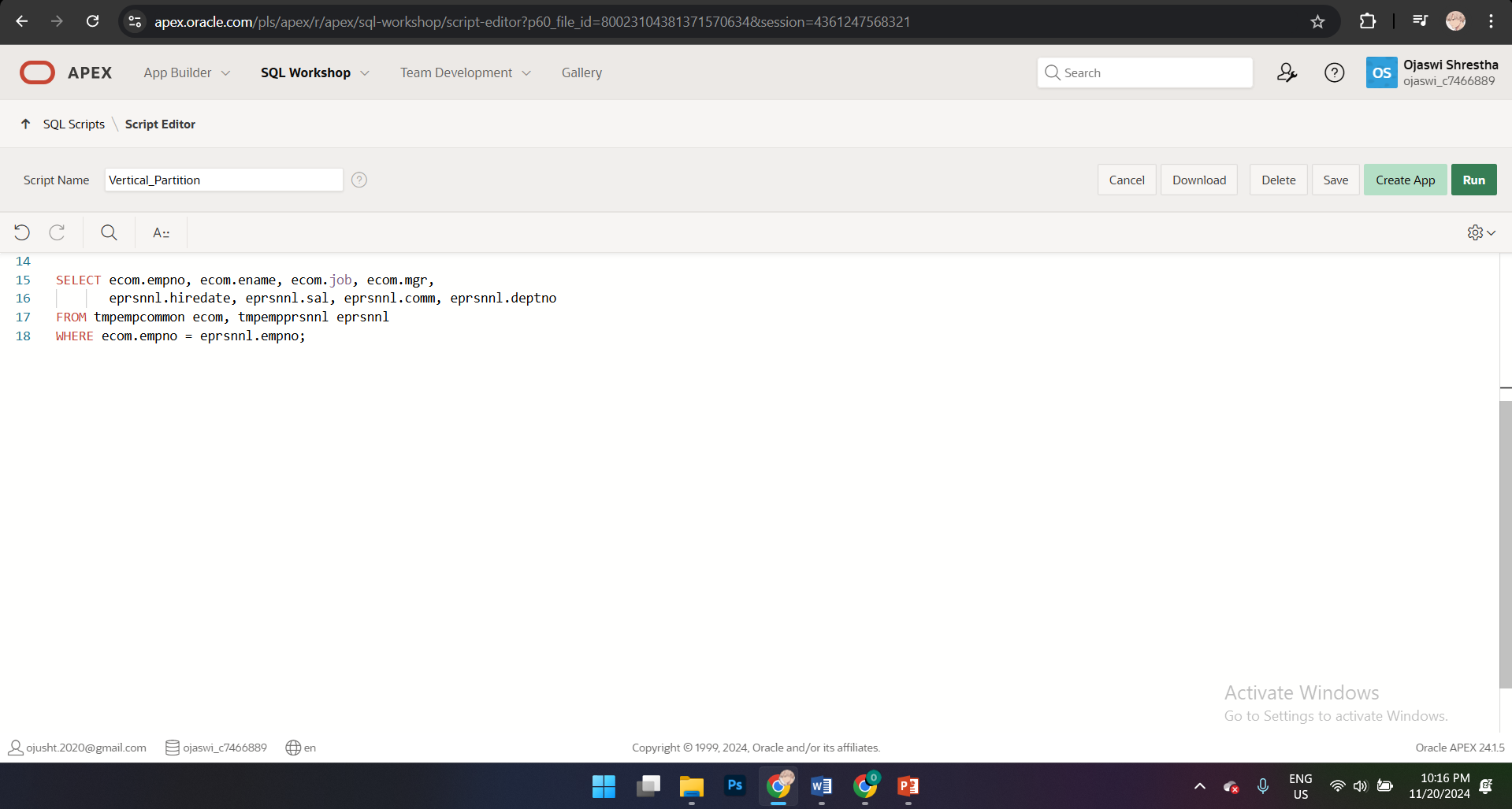


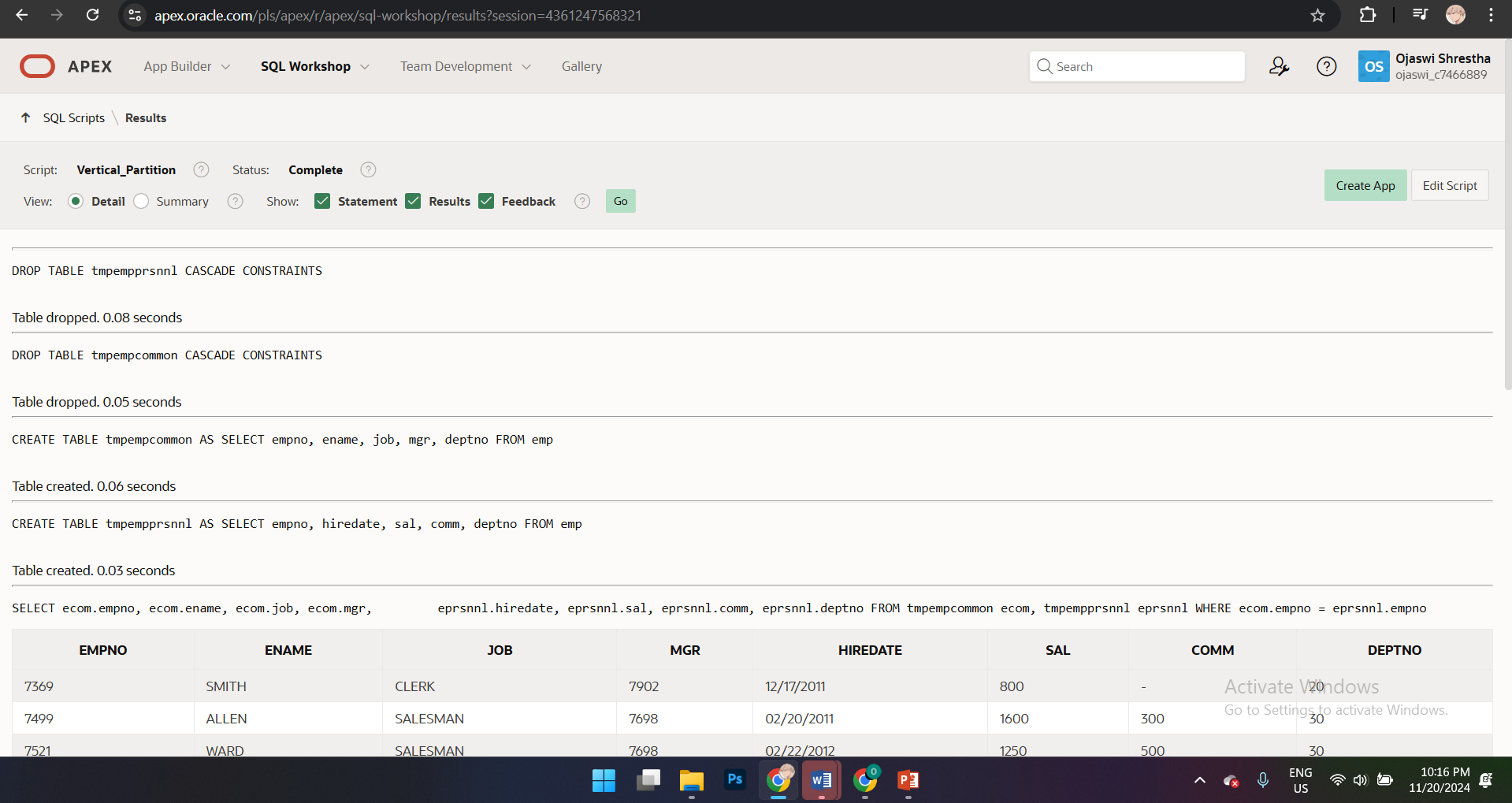


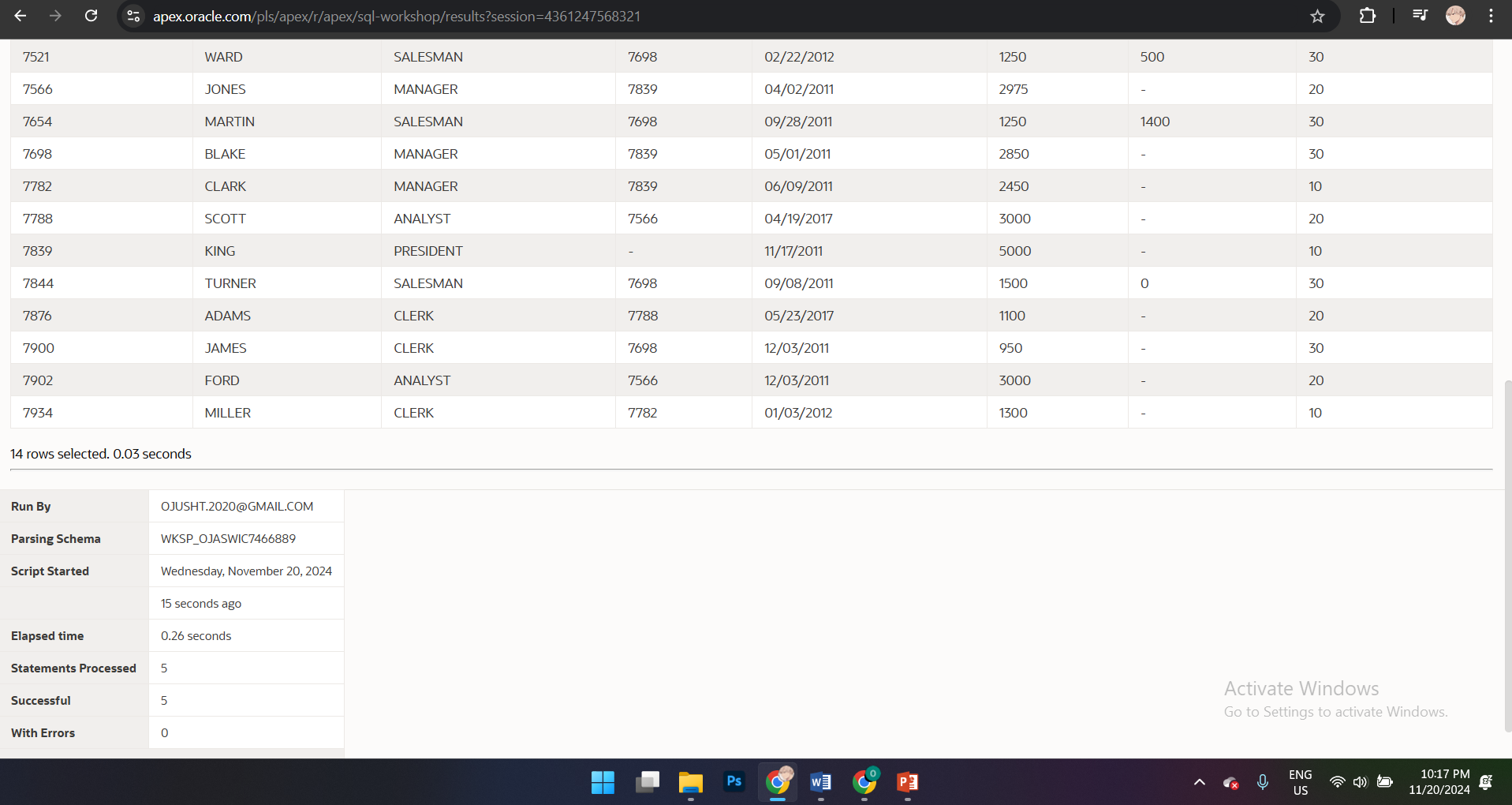




1. 2a) Now write SQL to select all details of all employees from tmpempprsnnl and tmpempcommon.







1. Do a combination of the above. There should be one tmpempprsnnl table containing all the sensitive data for all the employees and 4 other emp tables containing the general emp data (tmpempny, tmpempda etc). This is combined partitiosning.

