DATEBASE (SECTION)

2nd Task

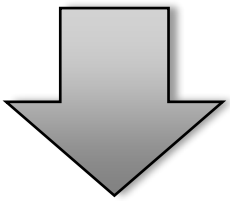
AHMED ELSAYED AHMED SNEED
ID: 4211142
A4

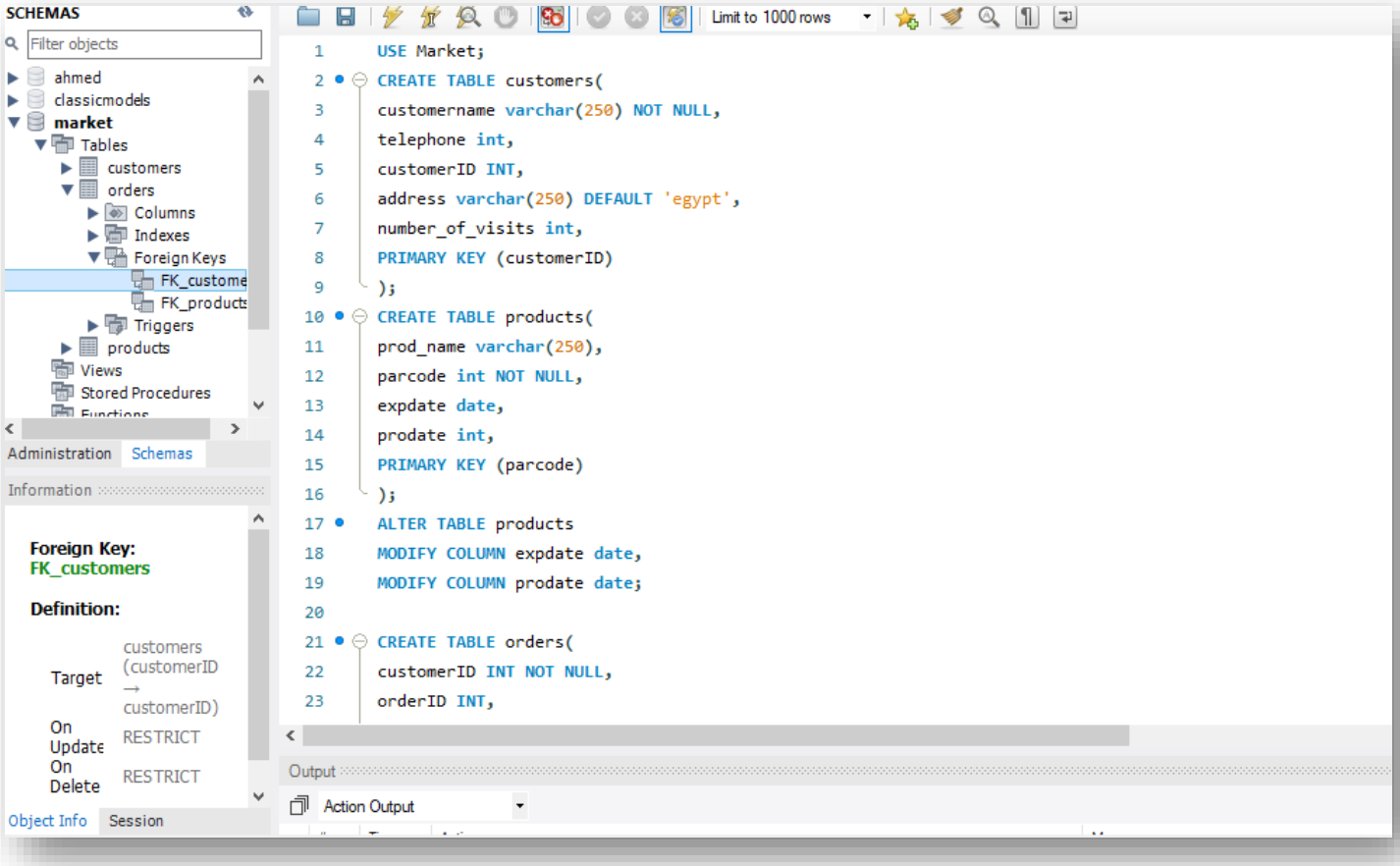
➢ Make a research on MySQL storage engines with a brief explanation for each type

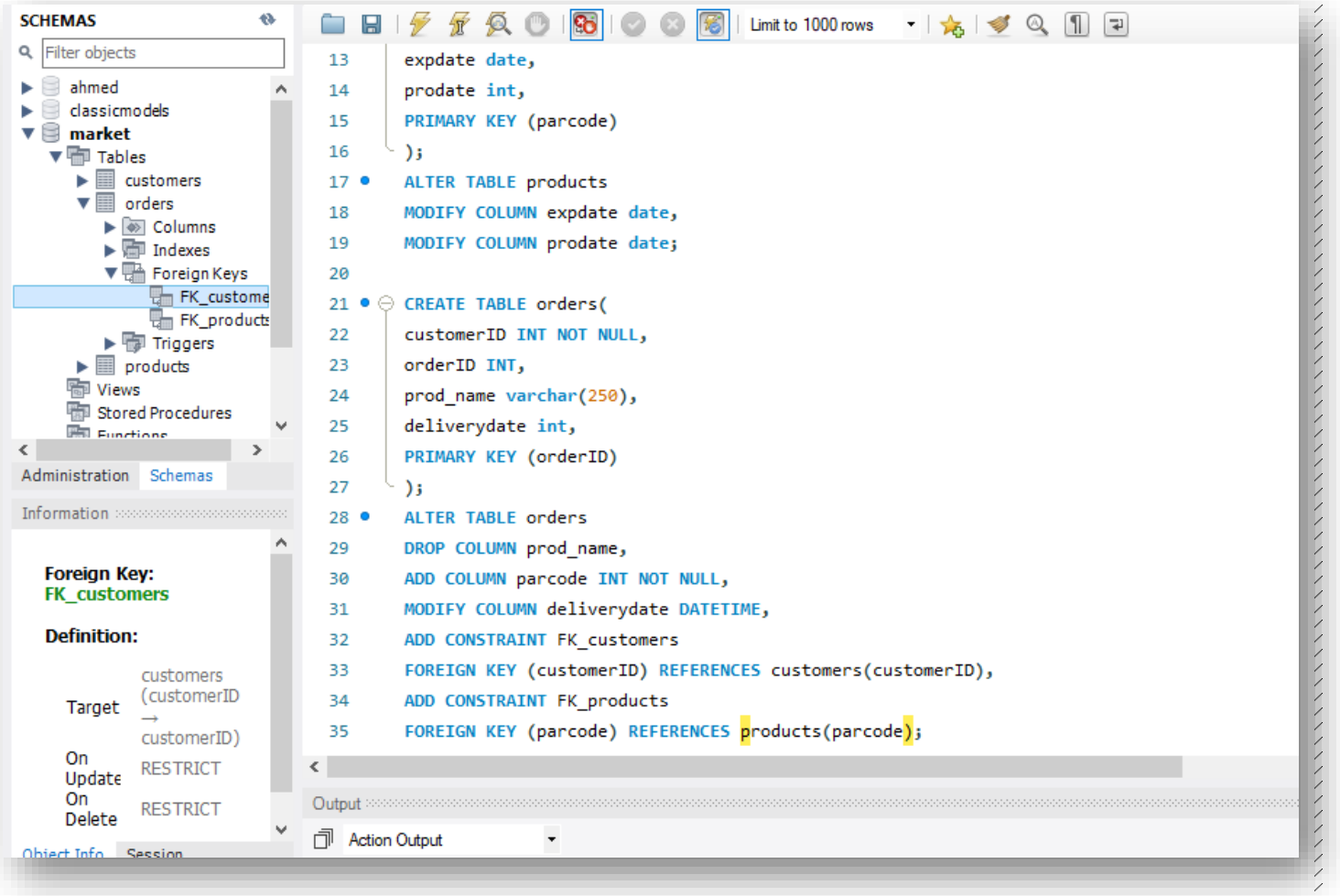
- ✓ **InnoDB:** the most widely used storage engine with transaction support. It is an ACID compliant storage engine. It supports row-level locking, crash recovery and multi-version concurrency control. It is the only engine which provides foreign key referential integrity constraint. Oracle recommends using InnoDB for tables except for specialized use cases.
- ✓ **MyISAM:** the original storage engine. It is a fast storage engine. It does not support transactions. provides table-level locking. It is used mostly in Web and data warehousing, manages non transactional tables, provides high-speed storage and retrieval, supports full text searching.
- ✓ **Memory:** It is the fastest engine. It provides table-level locking. It does not support transactions. Memory storage engine is ideal for creating temporary tables or quick lookups. The data is lost when the database is restarted.
- ✓ **CSV:** stores data in CSV files. It provides great flexibility because data in this format is easily integrated into other applications.
- ✓ **Merge:** operates on underlying MyISAM tables. Merge tables help manage large volumes of data more easily. It logically groups a series of identical MyISAM tables, and references them as one object. Good for data warehousing environments.
- ✓ **Archive:** optimised for high speed inserting. It compresses data as it is inserted. It does not support transactions. It is ideal for storing and retrieving large amounts of seldom referenced historical, archived data.
- ✓ **Federated:** offers the ability to separate MySQL servers to create one logical database from many physical servers. Queries on the local server are automatically executed on the remote tables. No data is stored on the local tables. It is good for distributed environments.
- ✓ **Blackhole:** accepts but does not store data. Retrievals always return an empty set. The functionality can be used in distributed database design where data is automatically replicated, but not stored locally. This storage engine can be used to perform performance tests or other testing.

Practical..

- Create a database (Market)
- Create tables in the image
- Create a primary key for every table
- Make relations between the tables using foreign keys






SCHEMAS
Filter objects
ahmed
classicmodels
market
Tables
customers
orders
Columns
Indexes
Foreign Keys
FK_custome
FK_products
Triggers
products
Views
Stored Procedures
Administration
Schemas
Information
Foreign Key:
FK_customers
Definition:
Target
customers (customerID → customerID)
On Update
RESTRICT
On Delete
RESTRICT
Object Info
Session
Limit to 1000 rows
13 expdate date,
14 prodate int,
15 PRIMARY KEY (parcode)
16);
17 ALTER TABLE products
18 MODIFY COLUMN expdate date,
19 MODIFY COLUMN prodate date;
20
21 CREATE TABLE orders(
22 customerID INT NOT NULL,
23 orderID INT,
24 prod_name varchar(250),
25 deliverydate int,
26 PRIMARY KEY (orderID)
27);
28 ALTER TABLE orders
29 DROP COLUMN prod_name,
30 ADD COLUMN parcode INT NOT NULL,
31 MODIFY COLUMN deliverydate DATETIME,
32 ADD CONSTRAINT FK_customers
33 FOREIGN KEY (customerID) REFERENCES customers(customerID),
34 ADD CONSTRAINT FK_products
35 FOREIGN KEY (parcode) REFERENCES products(parcode);
Output
Action Output