**מערכות קבצים ומסדי נתונים – פרויקט תכנות**

**שאילתות SQL**

|  |  |
| --- | --- |
| **מס' פעולה ותיאור** | **שאילתת SQL** |
| 1. הצגת כל הספרים הנמצאים במלאי | SELECT STOCK.book\_id, BOOK.book\_name, BOOK.author, STOCK.quantity  FROM  STOCK  inner join BOOK  on (STOCK.book\_id = BOOK.book\_id) |
| 1. הצגת כל ההזמנות (הפתוחות) לספרים | SELECT ORDERS.order\_id, ORDERS.order\_date, ORDER\_STATUS.order\_status\_name  ,BOOK.book\_id, BOOK.book\_name, BOOK.author, CUSTOMER.cust\_name, CUSTOMER.cust\_phone  FROM  ORDERS  inner join BOOK  on (ORDERS.book\_id = BOOK.book\_id)  inner join CUSTOMER  on (ORDERS.cust\_id = CUSTOMER.cust\_id)  inner join ORDER\_STATUS  on (ORDERS.order\_status\_id = ORDER\_STATUS.order\_status\_id)  where  ORDER\_STATUS.order\_status\_id <> 4 |
| 1. הצגת רשימת כל הלקוחות | SELECT \*  FROM  CUSTOMER  where  had\_purchased = 1 |
| 1. הצגת רשימת כל הספקים | SELECT \*  FROM  SUPPLIER |
| 1. הצגת כל העסקאות בטווח תאריכים נבחר | SELECT p.PURCHASE\_ID, p.purchase\_date, ps.purchase\_status\_name, p.purchase\_amount,  b.book\_name, c.cust\_name, e.emp\_name  FROM  PURCHASES p  inner join PURCHASE\_STATUS ps  on (p.purchase\_status\_id = ps.purchase\_status\_id)  inner join book b  on (p.book\_id = b.book\_id)  inner join CUSTOMER c  on (p.cust\_id = c.cust\_id)  inner join EMPLOYEE e  on (p.emp\_id = e.emp\_id)  WHERE  p.purchase\_date between ?? and ?? |
| 1. הצגת כל הספרים המוצעים בהנחה (גלובלית) | SELECT b.book\_id, b.book\_name, b.author, s.global\_discount  FROM  STOCK s  inner join book b  on (s.book\_id = b.book\_id)  WHERE  s.global\_discount > 0 |
| 1. בדיקה האם ספר X קיים כרגע במלאי החנות | use book\_store;  SELECT t.book\_name, t.author,  case when t.quantity is null then "not in stock"  when t.quantity = 0 then "not in stock"  else "in stock"  end is\_in\_stock  FROM (  SELECT b.book\_name, b.author, s.quantity  FROM  BOOK b  left join STOCK s  on (s.book\_id = b.book\_id)  WHERE  b.book\_name = X  OR b.author = X  ) t |
| 1. רשימת כל הספקים של ספר X | SELECT b.book\_name, b.author, s.\*  FROM  BOOK b  left join CAN\_SUPPLY cs  on (cs.book\_id = b.book\_id)  left join SUPPLIER s  on (cs.supp\_id = s.supp\_id)  WHERE  b.book\_name = X  or b.author = X |
| 1. כמה עותקים של ספר X נמכרו מאז Y | SELECT b.book\_name, b.author, count(distinct p.purchase\_id) purchases  FROM  BOOK b  left join PURCHASES p  on (p.book\_id = b.book\_id and p.purchase\_date > Y)  WHERE  b.book\_name = X  or b.author = X |
| 1. כמה ספרים בסה"כ מאז Y רכש לקוח Z | SELECT c.cust\_id, c.cust\_name, c.cust\_phone, count(distinct p.purchase\_id) purchases  FROM  CUSTOMER c  left join PURCHASES p  on (c.cust\_id = p.cust\_id and p.purchase\_date > Y)  WHERE  c.cust\_name = Z or c.cust\_id = Z or c.cust\_phone = Z  GROUP BY c.cust\_id, c.cust\_name, c.cust\_phone |
| 1. פרטי הלקוח שרכש הכי הרבה ספרים בחנות מאז Y | SELECT \*  FROM  (  SELECT c.cust\_id, c.cust\_name, c.cust\_phone, count(distinct p.purchase\_id) purchases  FROM  CUSTOMER c  join PURCHASES p  on (c.cust\_id = p.cust\_id)  WHERE  p.purchase\_date >Y  GROUP BY c.cust\_id, c.cust\_name, c.cust\_phone  ) cust\_max  join  (  select max(purchases) maximum  from (  SELECT c.cust\_id, c.cust\_name, c.cust\_phone, count(distinct p.purchase\_id) purchases  FROM  CUSTOMER c  join PURCHASES p  on (c.cust\_id = p.cust\_id)  WHERE  p.purchase\_date > Y  GROUP BY c.cust\_id, c.cust\_name, c.cust\_phone  ) t  ) all\_max  on (cust\_max.purchases = all\_max.maximum) |
| 1. פרטי הספק ממנו הוזמנו מאז Y הכי הרבה ספרים | SELECT \*  FROM  (  SELECT s.supp\_id, s.supp\_name, sum(sp.quantity) quantity  FROM  SUPPLIER s  join SUPPLIES sp  on (s.supp\_id = sp.supp\_id)  WHERE  sp.supply\_date > Y  GROUP BY s.supp\_id, s.supp\_name  ) a  join  (  select max(quantity) maximum  from (  SELECT s.supp\_id, s.supp\_name, sum(sp.quantity) quantity  FROM  SUPPLIER s  join SUPPLIES sp  on (s.supp\_id = sp.supp\_id)  WHERE  sp.supply\_date > Y  GROUP BY s.supp\_id, s.supp\_name  ) t  ) b  on (a.quantity = b.maximum) |
| 1. מספר ההזמנות לספרים שבוצעו בטווח תאריכים | SELECT count(order\_id)  FROM  ORDERS  WHERE  order\_date between ? and ? |
| 1. מספר כל ההזמנות לספרים בטווח תאריכים שהפכו לבסוף למכירות | SELECT count(order\_id)  FROM  ORDERS  WHERE  order\_status\_id = 4  and order\_date between ?? and ?? |
| 1. מה סה"כ ההנחה שקיבל לקוח Z מאז Y | SELECT sum(p.purchase\_amount\*p.cust\_discount/100) discount\_amount  FROM  PURCHASES p  join CUSTOMER c  on (p.cust\_id = c.cust\_id)  WHERE  p.cust\_discount > 0  and p.purchase\_date >= Y  and c.cust\_name = Z |
| 1. סך ההכנסות של החנות במהלך כל רבעון | SELECT purchase\_quarter, sum(amount\_after\_dis) amount  FROM (  SELECT p.purchase\_date,year(p.purchase\_date)\*100+quarter(p.purchase\_date) purchase\_quarter,  p.purchase\_amount, p.cust\_discount, p.purchase\_amount\*(100-p.cust\_discount)/100 amount\_after\_dis  FROM  PURCHASES p  WHERE  p.purchase\_status\_id = 1  ) t  GROUP BY purchase\_quarter  ORDER BY 1 |
| 1. כמה לקוחות חדשים התווספו מאז Y | SELECT count(cust\_id)  FROM  CUSTOMER  WHERE  cust\_creation\_date >= Y |
| 1. סך הרכישות בש"ח מספק בטווח תאריכים | SELECT sum(total)  FROM (  SELECT sp.book\_id, sp.quantity, cs.price, sp.quantity\*cs.price total  FROM  SUPPLIES sp  inner join SUPPLIER s  on (sp.supp\_id = s.supp\_id)  left join CAN\_SUPPLY cs  on (sp.supp\_id = cs.supp\_id and sp.book\_id = cs.book\_id)  WHERE  sp.supply\_date between ?? and ??  and s.supp\_id = W  ) t |
| 1. סך המכירות של מוכר בחנות בטווח תאריכים | SELECT count(distinct p.purchase\_id)  FROM  PURCHASES p  inner join EMPLOYEE e  on (p.emp\_id = e.emp\_id)  WHERE  p.purchase\_date between ?? and ??  and (e.emp\_id = W or e.emp\_name = W) |
| 1. רשימת 10 הספרים הנמכרים ביותר בטווח תאריכים | SELECT b.book\_id, b.book\_name, b.author, count(distinct p.purchase\_id) purchase\_count  FROM  PURCHASES p  inner join BOOK B  on (p.book\_id = b.book\_id)  WHERE  p.purcahse\_date between ?? and ??  group by b.book\_id, b.book\_name, b.author  ORDER BY count(distinct p.purchase\_id) DESC LIMIT 10 |