שאילתות

*מסמך זה מפרט את כלל השאילתות הנתנו בעבודת התכנות בפורמט הבא:

1. השאילתא שאנחנו מחפשים
2. קוד SQL למשיכת השאילתא לc++
3. הפונקציה המלאה בc++

שאילתא 1

האם ספר X נמצא במלאי?

SELECT COUNT(*) FROM book_instock INNER JOIN book_extension INNER JOIN book WHERE book_instock.book_extension_id = book_extension.book_extension_id AND book_extension.book_id = book.book_id AND book.book_name = 'input';

```
void AnarAndYanivBookShop::getBookFromStockByBookName()
{
    string str , input;
    int check_res;
    cout << "\n\tBook Name: ";
    cin.ignore();
    getline(cin, input);
    check_res = mysql.returnSqlQuery(str, "SELECT COUNT(*) FROM book_instock INNER JOIN book_extension
INNER JOIN book WHERE book_instock.book_extension_id = book_extension.book_extension_id AND
book_extension.book_id = book.book_id AND book.book_name = '" + input + "';");
    if (check_res &&  str != "0")
    {
        cout << "\n\tWe have " << str << " books in stock of '" + input + "'.\n";
    }
    else
        cout << "\n\tThe book '" + input + "' not exsist in the shop.\n\tTry to order it.\n";
}
```

SELECT COUNT(*) FROM book_instock INNER JOIN book_extension INNER JOIN book_author_relation INNER JOIN authors WHERE book_instock.book_extension_id = book_extension.book_extension_id AND book_extension.book_id = book_author_relation.book_id AND book_author_relation.author_id = authors.author_id AND first_name = 'input1' AND last_name = 'input2'

```
void AnarAndYanivBookShop::getBookFromStockByAuthorName()
{
    string str, first_name , last_name;
    int check_res;
    cout << "\n\tAuthor Name: ";
    //cin.ignore();
    cin >> first_name >> last_name;
    check_res = mysql.returnSqlQuery(str, "SELECT COUNT(*) FROM book_instock INNER JOIN book_extension
INNER JOIN book_author_relation INNER JOIN authors WHERE book_instock.book_extension_id =
book_extension.book_extension_id AND book_extension.book_id = book_author_relation.book_id AND
book_author_relation.author_id = authors.author_id AND first_name = '" + first_name + "' AND last_name
= '" + last_name + "';");
    if (check_res && str != "0")
        cout << "\n\tWe have " << str << " books in stock of the author '" + first_name + " " +
last_name + "'.\n";
    else
        cout << "\n\tWe not have books of the author '" + first_name + " " + last_name + "' not exsist
in the shop.\n\tTry to order it.\n";
}
```

שאילתא 2

מי הוא הלקוח הוותיק ביותר ?

```
SELECT first_name, last_name
FROM customer WHERE customer_id =
(SELECT MIN(customer_id) FROM customer)
```

```
void AnarAndYanivBookShop::whoIsTheFirstCustomerInTheShop()
{
    string str, input , first_name , last_name;
    int check_res;
    check_res = mysql.returnSqlQuery(str, "SELECT first_name , last_name FROM customer WHERE
customer_id = (SELECT MIN(customer_id) FROM customer)");
    if (check_res)
        cout << "\n\tThe oldest customer of the shop is '" << str << "'\n";
    else
        cout << "\n\tError. " << str;
}
```

שאילתא 3

מה הוא הספר הוותיק ביותר?

```
SELECT book_name FROM book_instock
INNER JOIN book_extension INNER JOIN book
WHERE book_instock.book_extension_id = book_extension.book_extension_id AND
book_extension.book_id = book.book_id
AND(SELECT MIN(date_in) FROM book_instock)
LIMIT 1
```

```
void AnarAndYanivBookShop::oldestBookInTheStock()
{
    string str;
    int check_res;
    check_res = mysql.returnSqlQuery(str, "SELECT book_name FROM book_instock INNER JOIN
book_extension INNER JOIN book WHERE book_instock.book_extension_id = book_extension.book_extension_id
AND book_extension.book_id = book.book_id AND(SELECT MIN(date_in) FROM book_instock) LIMIT 1");
    if (check_res)
        cout << "\n\tThe oldest book in the stock is: '" << str << "'" << endl;
    else
        cout << "\n\tError. " << str;
}
```

שאילתא 4

רשימת הזמנות הנוכחית

SELECT order_id, date_in, book_name , language , publisher_name ,publish_year , first_name, last_name, mobile_number
FROM orders INNER JOIN book_extension INNER JOIN book INNER JOIN customer
WHERE book_extension.book_extension_id = orders.book_extension_id AND book_extension.book_id = book.book_id AND customer.customer_id = orders.customer_id
ORDER BY date_in

```
void AnarAndYanivBookShop::ordersList()
{
    string str;
    int check_res;
    check_res = mysql.returnSqlQuery(str, "SELECT order_id, date_in, book_name , language ,
publisher_name , publish_year , first_name, last_name, mobile_number FROM orders INNER JOIN
book_extension INNER JOIN book INNER JOIN customer WHERE book_extension.book_extension_id =
orders.book_extension_id AND book_extension.book_id = book.book_id AND customer.customer_id =
orders.customer_id ORDER BY date_in");
    if (check_res)
    {
        cout << "\n" << str << endl;
    }
    else
        cout << "\n\tError. " << str;
}
```

שאילתא 5

כמה עותקים של ספר Y נמכרו על ידי החנות?

SELECT COUNT(*) FROM books_sales
INNER JOIN book_extension INNER JOIN book
WHERE books_sales.book_extension_id = book_extension.book_extension_id AND book_extension.book_id = book.book_id AND book.book_name = 'input'

```
void AnarAndYanivBookShop::howMuchOfBookHasBeenSold()
{
    string str , input;
    int check_res;
    cout << "\n\tBook name: ";
    cin.ignore();
    getline(cin, input);
    check_res = mysql.returnSqlQuery(str, "SELECT COUNT(*) FROM books_sales INNER JOIN book_extension
INNER JOIN book WHERE books_sales.book_extension_id = book_extension.book_extension_id AND
book_extension.book_id = book.book_id AND book.book_name = '" + input + "'");
    if (check_res)
        cout << "\n\tThe book '" << input << "' has been sold " << str << " times.\n";
    else
        cout << "\n\tError. " << str;
}
```

שאילתא 6

מי הסופר הכי נקרא בתווך תאריכים X עד Y.

SELECT first_name , last_name , COUNT(*)
FROM books_sales INNER JOIN book_extension INNER JOIN book_author_relation INNER JOIN authors
WHERE books_sales.book_extension_id = book_extension.book_extension_id AND book_extension.book_id = book_author_relation.book_id AND book_author_relation.author_id = authors.author_id AND sale_date BETWEEN 'input1' AND 'input2' GROUP BY first_name , last_name LIMIT 1;

```
void AnarAndYanivBookShop::whoIsTheMostReadAuthor()
{
    string str, first_date, second_date;
    int check_res;
    cout << "\n\tInsert first date in the following format YYYY-MM-DD: ";
    cin >> first_date;
    cout << "\n\tInsert second date in the following format YYYY-MM-DD: ";
    cin >> second_date;

    check_res = mysql.returnSqlQuery(str, "SELECT first_name , last_name , COUNT(*) FROM books_sales 
INNER JOIN book_extension INNER JOIN book_author_relation INNER JOIN authors WHERE 
books_sales.book_extension_id = book_extension.book_extension_id AND book_extension.book_id = 
book_author_relation.book_id AND book_author_relation.author_id = authors.author_id AND sale_date 
BETWEEN '" + first_date + "' AND '" + second_date + "' GROUP BY first_name , last_name LIMIT 1;");
    if (check_res)
    {
        string first_name = str.substr(0, str.find('\t'));
        str = str.substr(str.find('\t') + 1, str.size());
        string second_name = str.substr(0, str.find('\t'));
        string count = str.substr(str.find('\t') + 1, str.size());

        cout << "\n\tThe most read author between '" << first_date << "' & '" << second_date << "' is 
'" << first_name << " " << second_name << "' ( " << count << " times ).\n";
    }
    else
        cout << "\n\tError. " << str;
}
```

שאילתא 7

רשימת 3 הלקוחות שרכשו הכי הרבה ספרים לאורך השנים

SELECT first_name , last_name , SUM(counter) FROM (SELECT customer.customer_id AS id, first_name, last_name , COUNT(*) AS counter
FROM deal INNER JOIN customer INNER JOIN deal_books_relation WHERE deal.customer_id = customer.customer_id AND deal_books_relation.deal_id = deal.deal_id
GROUP BY customer.customer_id
UNION
SELECT customer.customer_id AS id, first_name , last_name , COUNT(*) FROM delivery INNER JOIN delivery_books_relation INNER JOIN deal INNER JOIN customer WHERE delivery.delivery_id = delivery_books_relation.delivery_id AND delivery.deal_id = deal.deal_id AND deal.customer_id = customer.customer_id GROUP BY customer.customer_id) AS complex GROUP BY id LIMIT 3

```
void AnarAndYanivBookShop::listOf3BestCustomers()
{
    string str;
    int check_res;
    check_res = mysql.returnSqlQuery(str, "SELECT first_name , last_name ,  SUM(counter) FROM (SELECT 
customer.customer_id AS id, first_name, last_name , COUNT(*) AS counter FROM deal INNER JOIN customer 
INNER JOIN deal_books_relation WHERE deal.customer_id = customer.customer_id AND 
deal_books_relation.deal_id = deal.deal_id GROUP BY customer.customer_id UNION SELECT 
customer.customer_id AS id,  first_name , last_name , COUNT(*) FROM delivery INNER JOIN 
delivery_books_relation INNER JOIN deal INNER JOIN customer WHERE delivery.delivery_id = 
delivery_books_relation.delivery_id AND delivery.deal_id = deal.deal_id AND deal.customer_id = 
customer.customer_id GROUP BY customer.customer_id) AS complex GROUP BY id LIMIT 3;");
    if (check_res)
        cout << "\n\tThe 3 customers who have read the most:\n\n" << str << endl;
    else
        cout << "\n\tError. " << str;
}
```

שאילתא 8

מי הוא הספר עם מספר התרגומים הגדול ביותר שקיים כרגע במלאי

SELECT book_name , COUNT(*) FROM book_extension INNER JOIN book WHERE book_extension.book_id = book.book_id AND book_extension.language IS NOT NULL GROUP BY book_name LIMIT 1

```
void AnarAndYanivBookShop::whoIsTheBookWithMostTranslations()
{
    string str;
    int check_res;
    check_res = mysql.returnSqlQuery(str, "SELECT book_name , COUNT(*) FROM book_extension INNER JOIN 
book WHERE book_extension.book_id = book.book_id AND book_extension.language IS NOT NULL GROUP BY 
book_name LIMIT 1");
    if (check_res)
    {
        string name = str.substr(0, str.find('\t'));
        string count = str.substr(str.find('\t') + 1, str.size());
        cout << "\n\tThe book with most translations is '" << name << "' ( " << count << " 
translations )." << endl;
    }
    else
        cout << "\n\tError. " << str;
}
```

שאילתא 9

היסטוריית רכישות של לקוח X :
אלו ספרים רכש, באלו תאריכים ומה המחיר ששילם עבור כל ספר

```
SELECT book_name, books_sales.price, sale_date  FROM deal INNER JOIN customer INNER JOIN
deal_books_relation INNER JOIN books_sales INNER JOIN book_extension INNER JOIN book
WHERE  deal.deal_id = deal_books_relation.deal_id AND deal.customer_id =
customer.customer_id AND books_sales.books_sale_id = deal_books_relation.book_sale_id AND
books_sales.book_extension_id = book_extension.book_extension_id AND book.book_id =
book_extension.book_id AND first_name = 'input1' AND last_name = 'input2'
UNION
SELECT book_name , books_sales.price , sale_date FROM delivery INNER JOIN
delivery_books_relation INNER JOIN deal INNER JOIN customer INNER JOIN books_sales INNER
JOIN book_extension INNER JOIN book WHERE delivery.deal_id = deal.deal_id AND
delivery_books_relation.delivery_id = delivery.delivery_id AND deal.customer_id =
customer.customer_id AND books_sales.books_sale_id = delivery_books_relation.book_sale_id
AND books_sales.book_extension_id = book_extension.book_extension_id AND book.book_id =
book_extension.book_id AND first_name = 'input1' AND last_name = 'input2'
```

```
void AnarAndYanivBookShop::customerHistory()
{
    string str , first_name , last_name;
    int check_res;
    cout << "\n\tEnter customer's full name: ";
    cin >> first_name >> last_name;
    check_res = mysql.returnSqlQuery(str, "SELECT book_name, books_sales.price, sale_date  FROM deal
INNER JOIN customer INNER JOIN deal_books_relation INNER JOIN books_sales INNER JOIN book_extension
INNER JOIN book WHERE  deal.deal_id = deal_books_relation.deal_id AND deal.customer_id =
customer.customer_id AND books_sales.books_sale_id = deal_books_relation.book_sale_id AND
books_sales.book_extension_id = book_extension.book_extension_id AND book.book_id =
book_extension.book_id AND first_name = '" + first_name + "' AND last_name = '" + last_name + "' UNION
SELECT book_name , books_sales.price , sale_date FROM delivery INNER JOIN delivery_books_relation
INNER JOIN deal INNER JOIN customer INNER JOIN books_sales INNER JOIN book_extension INNER JOIN book
WHERE delivery.deal_id = deal.deal_id AND delivery_books_relation.delivery_id = delivery.delivery_id
AND deal.customer_id = customer.customer_id AND books_sales.books_sale_id =
delivery_books_relation.book_sale_id AND books_sales.book_extension_id =
book_extension.book_extension_id AND book.book_id = book_extension.book_id AND first_name = '" +
first_name + "' AND last_name = '" + last_name + "';");
    if (check_res)
        cout << "\n\tCustomer's History:\n\n" << str << endl;
    else
        cout << "\n\tError. " << str;
}
```

שאילתא 10

לגבי אדם X : היסטוריית הזמנות (מה ומתי הזמין. מסודר לפי תאריכי הזמנה. ציון האם הספר אכן נמצא, ואם כן האם רכש)

SELECT book_name , date_in , isArrived , isBought FROM orders INNER JOIN book_extension INNER JOIN book INNER JOIN customer WHERE orders.book_extension_id = book_extension.book_extension_id AND book_extension.book_id = book.book_id AND orders.customer_id = customer.customer_id AND customer.first_name = 'input1' AND customer.last_name = 'input2' ORDER BY date_in

```
void AnarAndYanivBookShop::orderHistoryByCustomer()
{
    string str, first_name, last_name;
    int check_res;
    cout << "Enter customer's full name: ";
    cin >> first_name >> last_name;
    check_res = mysql.returnSqlQuery(str, "SELECT book_name , date_in , isArrived , isBought FROM 
orders INNER JOIN book_extension INNER JOIN book INNER JOIN customer WHERE orders.book_extension_id = 
book_extension.book_extension_id AND book_extension.book_id = book.book_id AND orders.customer_id = 
customer.customer_id AND customer.first_name = '" + first_name + "' AND customer.last_name = '" + 
last_name + "' ORDER BY date_in;");
    if (check_res)
        cout << "\n\tCustomer's order History:\n\n" << str << endl;
    else
        cout << "\n\tError. " << str;
}
```

שאילתא 11

חישוב עלות משלוח.

*The query depends on number of inputs by the user – the example shows 1 input

SELECT SUM(weight) FROM book_instock INNER JOIN book_extension WHERE book_instock.book_extension_id = book_extension.book_extension_id AND book_instock_id = 'input'

```
void AnarAndYanivBookShop::calculateDeliveryPriceBySelectedBooks()
{
    string str , input , check_str;
    vector<int> selectedBooks;
    int check_res , i = 0;
    cout << "\n\tEnter books from stock by their id('book_instock_id'):\n";
    while (1)
    {
        cout << "\n\tInsert the serial number of the book in the shop (to finish Enter 'f'): ";
        cin >> input;
        if (input == "f" || input == "F")
            break;
        if (!i)
        {
            check_str += "book_instock_id = " + input;
            i++;
        }
        else
            check_str += " OR book_instock_id = " + input;
    }
    check_res = mysql.returnSqlQuery(str, "SELECT SUM(weight) FROM book_instock INNER JOIN 
book_extension WHERE book_instock.book_extension_id = book_extension.book_extension_id AND(" + 
check_str + ");");
    if (check_res)
    {
        int option;
        float sum = 0;
        sscanf_s(str.c_str(), "%f", &sum);
        cout << "\n\tChoose one of the delivery options:\n\n\t1.Israel Post - Registered 
mail\n\t2.Israel Post - Courier service\n\t3.Israel Post - Fast Courier service\n\t4.Xpress - Take 
away\n\t5.Xpress - Courier service\n\n\t";
        cin >> option;
        if (option == 1)
        {
            sum *= ISRAEL_POST_KG_PRICE;
            sum += ISRAEL_POST_SLOW_PRICE;
        }



        if (option == 2)
        {
            sum *= ISRAEL_POST_KG_PRICE;
            sum += ISRAEL_POST_AVG_PRICE;
        }
        if (option == 3)
        {
            sum *= ISRAEL_POST_KG_PRICE;
            sum += ISRAEL_POST_SLOW_PRICE;
        }

        if (option == 4)
        {
            sum *= XPRESS_KG_PRICE;
            sum += XPRESS_TA_PRICE;
        }
        if (option == 5)
        {
            sum *= XPRESS_KG_PRICE;
            sum += XPRESS_DELIVERY_PRICE;
        }
        cout << "\n\tThe price of this delivery is '" << sum << "'" << endl;
    }
    else
        cout << "\n\tError. " << str;
}
```

שאילתא 12

האם לקוח X פיצל, אי פעם, רכישת ספרים למספר משלוחים ואם כן מה הם נתוני המשלוחים

SELECT tracking_id, delivery_status, courier_company, delivery_type, deal.date_in FROM delivery INNER JOIN deal WHERE delivery.deal_id = deal.deal_id AND delivery.deal_id = (SELECT deal.deal_id FROM delivery INNER JOIN deal INNER JOIN customer WHERE delivery.deal_id = deal.deal_id AND customer.customer_id = deal.customer_id AND first_name = 'input1' AND last_name = 'input2' HAVING COUNT(deal.deal_id) > 1)

```
void AnarAndYanivBookShop::getInfoIfCustomerEverSplitDealToFewDeliveries()
{
    string str , first_name , last_name;
    int check_res;
    cout << "\n\tEnter Customer full name: ";
    cin >> first_name >> last_name;
    check_res = mysql.returnSqlQuery(str, "SELECT tracking_id, delivery_status, courier_company,
delivery_type, deal.date_in FROM delivery INNER JOIN deal WHERE delivery.deal_id = deal.deal_id AND
delivery.deal_id = (SELECT deal.deal_id FROM delivery INNER JOIN deal INNER JOIN customer WHERE
delivery.deal_id = deal.deal_id AND customer.customer_id = deal.customer_id AND first_name = '" +
first_name + "' AND last_name = '" + last_name + "' HAVING COUNT(deal.deal_id) > 1);");
    if (check_res)
        cout << "\n\t'" + first_name + " " + last_name + "' had splitted deals:\n\n" << str << endl;
    else
        cout << "\n\t'" + first_name + " " + last_name + "' not had splitted deals.\n" << endl;
}
```

שאילתא 13

מה הוא הסטטוס הנוכחי של משלוח מסוים

SELECT delivery_status FROM delivery WHERE delivery_id = input

```
void AnarAndYanivBookShop::getDeliveryStatus()
{
    string str, id;
    int check_res;
    cout << "\n\tInsert delivery's serial number: ";
    cin >> id;
    check_res = mysql.returnSqlQuery(str, "SELECT delivery_status FROM delivery WHERE delivery_id = "
+ id + " ;");
    if (check_res)
        cout << "\n\tThe status is '" << str << "'" << endl;
    else
        cout << "\n\tError. " << str;
}
```

שאילתא 14

מה סכום המשלוחים שבוצעו על ידי חברת Xpress בחודש מסוים

SELECT COUNT(*) FROM delivery INNER JOIN deal WHERE delivery.deal_id = deal.deal_id AND courier_company = 'Xpress' AND date_in BETWEEN 'input-01' AND 'input-31'

```
void AnarAndYanivBookShop::getNumberOfDeliveriesThatDoneByXpressInSpecificMonth()
{
    string str, input;
    int check_res;
    cout << "\n\tInsert month and year by following format YYYY-MM: ";
    cin >> input;
    check_res = mysql.returnSqlQuery(str, "SELECT COUNT(*) FROM delivery INNER JOIN deal WHERE 
delivery.deal_id = deal.deal_id AND courier_company = 'Xpress' AND date_in BETWEEN '" + input + "-01' 
AND '" + input + "-31' ;");
    if (check_res)
        cout << "\n\tXpress made "<< str << " deliveries at '" << input << "'" << endl;
    else
        cout << "Error. " << str;
}
```

שאילתא 15

סך הכסף שהועבר לחשבון החנות באמצעות אפליקציית Bit בחודש מסוים

SELECT SUM(price) FROM deal WHERE payment_type = 'Bit' AND date_in BETWEEN 'input-01' AND 'input-31'

```
void AnarAndYanivBookShop::getSumOfBitAppIncomeInSpecificMonth()
{
    string str, input;
    int check_res;
    cout << "\n\tInsert month and year by following format YYYY-MM: ";
    cin >> input;
    check_res = mysql.returnSqlQuery(str, "SELECT SUM(price) FROM deal WHERE payment_type = 'Bit' AND 
date_in BETWEEN '" + input + "-01' AND '" + input + "-31';");
    if (check_res)
        cout << "\n\tPaid in " + input + " by the BIT app '" + str + "' Shekels."<< endl;
    else
        cout << "\n\tError. " << str;
}
```

שאילתא 16

מהן העסקאות שבוצעו במהלך 12 החודשים האחרונים, ואשר הניבו רווח גדול יותר מרווח העסקאות הממוצע ב- 12 החודשים האלו

SELECT customer.first_name, customer.last_name, date_in, price, payment_type, employee.first_name, employee.last_name FROM deal INNER JOIN customer INNER JOIN employee WHERE deal.employee_id = employee.employee_id AND deal.customer_id = customer.customer_id AND price > (SELECT AVG(price) FROM deal WHERE date_in >= NOW() - INTERVAL 1 YEAR)

```
void AnarAndYanivBookShop::getDealsFromPastYearAndMoreProfitableThanLastYearAVG()
{
    string str;
    int check_res;

    check_res = mysql.returnSqlQuery(str, "SELECT customer.first_name, customer.last_name, date_in,
price, payment_type, employee.first_name, employee.last_name FROM deal INNER JOIN customer INNER JOIN
employee WHERE deal.employee_id = employee.employee_id AND deal.customer_id = customer.customer_id AND
price > (SELECT AVG(price) FROM deal WHERE date_in >= NOW() - INTERVAL 1 YEAR);");
    if (check_res)
        cout << "\n\tThe deals from past year that are more profitable than the avarage is:\n\n" + str
<< endl;
    else
        cout << "\n\tError. " << str;

}
```

שאילתא 17

כמה משלוחים בוצעו במהלך 12 החודשים האחרונים באמצעות דואר ישראל, וכמה בוצעו באמצעות חברת Xpress

SELECT COUNT(*) FROM delivery WHERE courier_company = 'Israel Post'
SELECT COUNT(*) FROM delivery WHERE courier_company = 'Xpress'

```
void AnarAndYanivBookShop::howManyDeliveriesDeliveredWithEachCompany()
{
    string str;
    int check_res , israel_post, xpress;
    check_res = mysql.returnSqlQuery(str, "SELECT COUNT(*) FROM delivery WHERE courier_company =
'Israel Post'");
    sscanf_s(str.c_str(), "%d", &israel_post);
    str = "";
    check_res = mysql.returnSqlQuery(str, "SELECT COUNT(*) FROM delivery WHERE courier_company =
'Xpress'");
    sscanf_s(str.c_str(), "%d", &xpress);
    cout << "\n\tAt past 12 months delivered by Israel Post " << israel_post << " deliveries and by
Xpress " << xpress << " deliveries." << endl;
}
```

שאילתא 18

נתונים על כל המשלוחים שבוצעו, אי פעם, וכללו לפחות 2 מהדורות שונות של אותו הספר.

SELECT delivery.delivery_id, delivery.delivery_type, delivery.delivery_status, delivery.deal_id FROM delivery INNER JOIN delivery_books_relation INNER JOIN books_sales INNER JOIN book_extension WHERE delivery.delivery_id = delivery_books_relation.delivery_id AND delivery_books_relation.book_sale_id = books_sales.books_sale_id AND books_sales.book_extension_id = book_extension.book_extension_id GROUP BY book_id HAVING COUNT(book_extension.book_extension_id) > 1;

```
void AnarAndYanivBookShop::allDeliveriesWithTwoDiffrentBookEditions()
{
    string str;
    int check_res;
    check_res = mysql.returnSqlQuery(str, "SELECT delivery.delivery_id, delivery.delivery_type, 
delivery.delivery_status, delivery.deal_id FROM delivery INNER JOIN delivery_books_relation INNER JOIN 
books_sales INNER JOIN book_extension WHERE delivery.delivery_id = delivery_books_relation.delivery_id 
AND delivery_books_relation.book_sale_id = books_sales.books_sale_id AND books_sales.book_extension_id 
= book_extension.book_extension_id GROUP BY book_id HAVING COUNT(book_extension.book_extension_id) > 
1;");
    if (check_res)
        cout << "Those are the deliveries that has two diffrent editions of the same book: " + str << 
endl;
    else
        cout << "Error." << str;
}
```

שאילתא 19

נתונים על כל הלקוחות שרכשו בעבר, מתי שהוא, לפחות ספר אחד מהחנות, ושלא ביצעו שום רכישה במהלך 24 החודשים האחרונים

SELECT customer.first_name, customer.last_name, customer.phone_number, customer.mobile_number FROM customer INNER JOIN deal WHERE deal.customer_id = customer.customer_id AND deal.date_in <= NOW() - INTERVAL 24 MONTH GROUP BY customer.customer_id

```
void  AnarAndYanivBookShop::customersDidntBuyBookInLast24Month()
{
    string str;
    int check_res;
    check_res = mysql.returnSqlQuery(str, "SELECT customer.first_name, customer.last_name, 
customer.phone_number, customer.mobile_number FROM customer INNER JOIN deal WHERE deal.customer_id = 
customer.customer_id AND deal.date_in <= NOW() - INTERVAL 24 MONTH GROUP BY customer.customer_id;");
    if (check_res)
        cout << "\n\tThese are the customers that didn't buy any book in the last 24 months:\n" + str 
<< endl;
    else
        cout << "\n\tThere is not customers that didn't buy any book in the last 24 months" << str;
}
```

שאילתא 20

רשימת הלקוחות שביצעו הזמנות, הספרים שהזמינו הגיעו לחנות, החנות יצרה איתם קשר ליידע אותם על זמינות הספר. הקשר נוצר לפני 14 ימים, והלקוחות עדיין לא רכשו הספר.

SELECT first_name , last_name , phone_number , mobile_number FROM orders INNER JOIN customer WHERE orders.customer_id = customer.customer_id AND isArrived = 1 AND date_arrived <= NOW() - INTERVAL 14 DAY AND isBought = 0

```
void  AnarAndYanivBookShop::getOrdersThatArrivedBefore14DaysAndNotBoughtYet()
{
    string str;
    int check_res;
    check_res = mysql.returnSqlQuery(str, "SELECT first_name , last_name , phone_number , 
mobile_number FROM orders INNER JOIN customer WHERE orders.customer_id = customer.customer_id AND 
isArrived = 1 AND date_arrived <= NOW() - INTERVAL 14 DAY AND isBought = 0");
    if (check_res)
        cout << "\n\tThese are the orders that arrived to the shop, the customer has been updated 
about that in 14 days before and not come/buy it yet:\n" + str << endl;
    else
        cout << "\n\tWe not have orders that arrived to the shop, the customer has been updated about 
that in 14 days before and not come/buy it yet" << str;
}
```

שאילתא 21

מספר הספרים במחסן בחתך חודשי.

SELECT YEAR(date_in), MONTH(date_in), COUNT(*) FROM book_instock GROUP BY MONTH(date_in), YEAR(date_in) ORDER BY date_in

```
void AnarAndYanivBookShop::getNumberOfBooksInShopByMonthlyCut()
{
    string str;
    int check_res;
    check_res = mysql.returnSqlQuery(str, "SELECT YEAR(date_in), MONTH(date_in), COUNT(*) FROM 
book_instock GROUP BY MONTH(date_in), YEAR(date_in) ORDER BY date_in;");
    if (check_res)
    {
        cout << "\n\tList of number of books that arrived to the shop order by year and month:\n\n" << 
str << endl;
    }
    else
        cout << "\n\tThere is not customers that didn't buy any book in the last 24 months" << str;
}
```

שאילתא 22

כמה ספרים רכשה החנות בין תאריך D1 לתאריך, D2 ומה היה סך התשלום עבורם

SELECT COUNT(*) , SUM(price) FROM book_instock WHERE date_in BETWEEN 'input1' AND 'input2'

```
void AnarAndYanivBookShop::howManyBooksTheStoreBoughtBetweenDatesAndThePrice()
{
    string str, date1, date2;
    int check_res;
    cout << "Insert fisrt date by following format YYYY-MM-DD:" << endl;
    cin >> date1;
    cout << "Insert second date by following format YYYY-MM-DD:" << endl;
    cin >> date2;
    check_res = mysql.returnSqlQuery(str, "SELECT COUNT(*) , SUM(price) FROM book_instock WHERE 
date_in BETWEEN '" + date1 + "' AND '" + date2 + "';");
    if (check_res)
        cout << "The number of books that the store bought between the dates " + date1 + " and " + 
date2 + " and their cost: " + str << endl;
    else
        cout << "Error." << str;
}
```

שאילתא 23

רווח החנות ממכירות בחודש מסוים

SELECT SUM(books_sales.price) - SUM(book_instock.price) FROM books_sales INNER JOIN book_instock WHERE MONTH(books_sales.sale_date) = input1 AND MONTH(book_instock.date_in) = input1 AND YEAR(books_sales.sale_date) = input2 AND YEAR(book_instock.date_in) = input2

```
void AnarAndYanivBookShop::storesProfitOnSpecificMonth()
{
    string str;
    int month, year, check_res;
    cout << "Insert The month you want to know store's profit: " << endl;
    cin >> month;
    cout << "Insert the year: " << endl;
    cin >> year;
    check_res = mysql.returnSqlQuery(str, "select SUM(books_sales.price) - SUM(book_instock.price) 
from books_sales inner join book_instock where MONTH(books_sales.sale_date) =  " + to_string(month) + 
" and MONTH(book_instock.date_in) = " + to_string(month) + " and YEAR(books_sales.sale_date) = " + 
to_string(year) + " and YEAR(book_instock.date_in) = " + to_string(year));
    if (check_res)
        cout << "Store's profit on " + to_string(month) + "--" + to_string(year) + ": " + str << endl;
    else
        cout << "Error." << str;
}
```

שאילתא 24

ממוצע עסקאות שנתי בחתך חודשי

SELECT SUM(price) FROM deal WHERE date_in >= NOW() - INTERVAL 1 YEAR

```
void AnarAndYanivBookShop::getPastYearDealsAVGbyMonthlyCut()
{
    string str;
    int check_res;
    check_res = mysql.returnSqlQuery(str, "SELECT SUM(price) FROM deal WHERE date_in >= NOW() -
INTERVAL 1 YEAR;");
    if (check_res)
    {
        float avg;
        sscanf_s(str.c_str(), "%f", &avg);
        avg /= 12;
        cout << "\n\tThe average of deals of the past year is '" + to_string(avg) + "'" << endl;
    }
}
```

שאילתא 25

המשכורת (ברוטו) של עובד Z בחודש מסוים

SELECT monthly_hours, salary FROM employee INNER JOIN employees_hours WHERE employee.employee_id = employees_hours.employee_id AND employee.first_name = 'input1' AND employee.last_name = 'input2' AND month = 'input3-01'

```
void AnarAndYanivBookShop::getSalaryOfEmployeeByMonth()
{
    string str , month , first_name , last_name;
    int check_res;
    cout << "\n\tInsert employee's full name : ";
    cin >> first_name >> last_name;
    cout << "\n\tInsert month in following XXXX-XX: ";
    cin >> month;
    check_res = mysql.returnSqlQuery(str, "SELECT monthly_hours, salary FROM employee INNER JOIN
employees_hours WHERE employee.employee_id = employees_hours.employee_id AND employee.first_name = '"
+ first_name + "' AND employee.last_name = '" + last_name + "' AND month = '" + month + "-01';");
    if(check_res)
    {
        double hours, salary;
        sscanf_s(str.c_str(), "%lf %lf", &hours, &salary);
        salary *= hours;
        cout << "\n\tThe salary of '" + first_name + " "  + last_name +  "' at month '" + month + "'
was " + to_string(salary) << endl;
    }
    else
        cout << "\n\tError." << str;
}
```

שאילתא 26

מי המוכר עם הכי הרבה עסקאות בחודש X

SELECT COUNT(*), SUM(deal.price) FROM deal WHERE deal.date_in BETWEEN 'input1' AND 'input2'

```cpp
void AnarAndYanivBookShop::sellerWithTheMostSalesPerMonth()
{
    string str, input;
    int check_res;
    cout << "\n\tInsert the month you want to know which seller had the most sales: " << endl;
    cin >> input;
    check_res = mysql.returnSqlQuery(str, "select first_name, last_name from deal inner join employee 
where employee.employee_id = deal.employee_id and MONTH(deal.date_in) = " + input + " group by 
employee.employee_id LIMIT 1;");
    if (check_res)
        cout << "\n\tThe seller with the most sales on month " + input + "is: " + str << endl;
    else
        cout << "\n\tError." << str;
}
```