**Вопрос 1** **Выберите верные утверждения для LINQ:**

Ключевое слово var может быть использовано, как в контексте LINQ, так и в любом другом месте программы, где это имеет смысл делать

Ключевое слово var создано только для использования в контексте LINQ

При использовании var реальный тип данных определяется на этапе исполнения программы

При создании LINQ запросов лучше не использовать ключевое слово var

При использовании var реальный тип данных определяется на этапе компиляции программы

**Ответ:** Ключевое слово var может быть использовано, как в контексте LINQ, так и в любом другом месте программы, где это имеет смысл делать   
При использовании var реальный тип данных определяется на этапе компиляции программы

**Вопрос 2** **Выберите верные утверждения относительно данных LINQ запросов:**

**from book in Books**

**where book.Genre.Equals("Comedy") && book.Pages > 100**

**select book**

**Books.Where(s=>s.Genre == "Comedy"&&s.Pages>100)**

Оба запроса не содержат синтаксических ошибок

Оба запроса решают одну и ту же задачу

Запросы решают разные задачи

Оба запроса содержат синтаксические ошибки

**Ответ:** Оба запроса не содержат синтаксических ошибок   
Оба запроса решают одну и ту же задачу

**Вопрос 3** **Выберите верное утверждение для LINQ:**

C помощью select нельзя выбрать два и более столбцов

where используется для группировки данных по критерию, указанному в нем

Синтаксис лямбд (синтаксис методов) и синтаксис запросов полностью взаимозаменяемы

Структура LINQ запроса не является жесткой. Запрос может начинаться, как с select, так и с where

let - ключевое слово используемое в синтаксисе запросов

**Ответ:** let - ключевое слово используемое в синтаксисе запросов

**Вопрос 4** **Выберите LINQ запрос, который является взаимозаменяемым относительно исходного:**

**Customers.GroupBy(d => d.Age).Select(**

**r => new{**

**ResultAge = r.Key,**

**MaxIncome = r.Max(l => l.Income),**

**}**

**)**

from customer in Customers

select new{

ResultAge = customer.Age,

MaxIncome = customerGroup.Max(i => i.Income)

}

from customer in Customers

group customer by customer.Age

select new{

ResultAge = customer.Key,

MaxIncome = customer.Max(i => i.Income)

}

from customer in Customers

group customer by customer.Age in customerGroup

select new{

ResultAge = customerGroup.Key,

MaxIncome = customerGroup. Max(i => i.Income)

}

from customer in Customers

group customer by customer.Age into customerGroup

select new{

ResultAge = customerGroup.Key,

MaxIncome = customerGroup.Max(i => i.Income)

}

**Ответ:** from customer in Customers   
group customer by customer.Age into customerGroup   
select new{   
 ResultAge = customerGroup.Key,   
 MaxIncome = customerGroup.Max(i => i.Income)   
}

**Вопрос 5** **Выберите верные утверждения относительно данного LINQ запроса:**

**from customer in Customers**

**where customer.Age>21 and customer.Age<32**

**select customer**

Запрос LINQ содержит синтаксическую ошибку

Расположение select в конце запроса является правильным с точки зрения LINQ

Запрос LINQ является абсолютно верным по синтаксису

Расположение select в конце запроса является синтаксической ошибкой

**Ответ:** Запрос LINQ содержит синтаксическую ошибку   
Расположение select в конце запроса является правильным с точки зрения LINQ

**Вопрос 6** **Какое из этих слов не является ключевым словом для LINQ?**

into

why

select

where

from

**Ответ:** why

**Вопрос 7** **Выберите методы для агрегирования в LINQ:**

Count

Min

Max

Avg

**Ответ:** Count   
Min   
Max

**Вопрос 8** **Выберите верное утверждение относительно данного LINQ запроса:**

**from customer in Customers**

**where customer.Lastname like '%K%'**

**select customer**

Запрос получает всех покупателей, чья фамилия начинается на K

В запросе есть ошибка синтаксиса. Вместо использования like, нужно использовать метод StartsWith

В select можно указать только тот идентификатор, который был использован во from. Если во from customer, то и в select только customer

Если указать в select, например, customer.Lastname, то получим синтаксическую ошибку LINQ

Конструкция like и вызов метода StartsWith абсолютно взаимозаменяемы

**Ответ:** В запросе есть ошибка синтаксиса. Вместо использования like, нужно использовать метод StartsWith

**Вопрос 9** **Выберите верные утверждения относительно данного LINQ запроса:**

**from book in Books**

**select new {**

**book.Title, book.Year**

**}**

Запрос LINQ содержит ошибку синтаксиса. Нужно поменять местами from и select

Запрос LINQ является абсолютно верным по синтаксису

new используется, когда необходимо создать проекцию. Например, вернуть два и более полей

Запрос LINQ содержит ошибку синтаксиса. Для её исправления вместо

select new {

book.Title, book.Year

}

нужно написать

select book.Title, book.Year

**Ответ:** Запрос LINQ является абсолютно верным по синтаксису   
new используется, когда необходимо создать проекцию. Например, вернуть два и более полей

**Вопрос 10** **Выберите верное утверждение относительно into:**

into используется в orderby

into используется в where

into используется в select

into используется в group

**Ответ:** into используется в group