



•



✓ Plan

#### 2023-09-11

- 1. Homework:
- area of the cube, triangle, circle;
- 2. New theme: compare operations. ternary operator. Conditional operator we teach the program to make decisions. See the theory section.
- 3. Practice:
- Task 1: Checking for a positive or negative number. Checking for an even or odd integer. Age verification. If the user is 18 years of age or less, then report "You need to come with your guardian," otherwise say, "You are allowed to act independently.
- Task 2: Declare two integer variables a and b and assign them arbitrary natural number values. Create a third variable res and print its value to the screen like this: If a = b, then res will be equal to (a+b) \* 5 and if not, then (a b) \* 5.
- Task 3. Write a method that returns the maximum of two integers, use the ternary operator. Add a method that returns the maximum of three integers. The numbers are set in the program.
- Task 4: Implement a method that determines whether one integer is divisible by another or not.
- Task 5: Develop an application for solving a quadratic equation using the coefficients of the quadratic trinomial a, b and c. Calculate the discriminant using the formula: d = bb 4ac If d > 0, then x1 = (-b + Math.sqrt(d))/(2a), x2 = (-b Math.sqrt(d))/(2a). If d = 0, then x = -b / (2a). If d < 0, then report that there are no roots.</li>



AIT LMS

- 1. Homework:
- площадь куба, треугольника, круга;
- 2. Новая тема: Compare operations. Ternary operator. Условный оператор учим программу принимать решения. См. раздел теория.
- 3. Практика:
- Задание 1: Проверка положительное или отрицательное число. Проверка четное или нечетное целое число. Проверка возраста. Если пользователю 18 лет и менее, то сообщить "Вам надо прийти вместе с вашим опекуном", иначе сообщить "Вам разрешено действовать самостоятельно.
- Задание 2: Объявите две переменные целого типа а и b и присвойте им произвольные значения натуральных чисел. Создайте третью переменную res и выведите ее значение на экран таким образом: Если a = b, то res будет равно (a+b) \* 5 а если нет, то (a b) \* 5.
- Задание 3. Написать метод, возвращающий максимальное из двух целых чисел, использовать тернарный оператор. Добавить метод, который возвращает максимальное из трех целых чисел. Числа задаются в программе.
- Задание 4: Реализовать метод, определяющий, делится ли одно целое число на другое нацело или нет.
- Задание 5: Разработать аппликацию для решения квадратного уравнения по коэффициентам квадратного трехчлена a, b и c. Дискриминант вычислить по формуле: d = bb 4ac Если d > 0, mo x1 = (-b + Math.sqrt(d))/(2a), x2 = (-b Math.sqrt(d))/(2a). Если d = 0, mo x = -b / (2a). Если d < 0, то сообщить, что корней нет.</li>

∨ Theory

# **Conditional operator**

#### if() statement

The most versatile way to organize the execution of some part of the code depending on the condition, is to use the conditional operator if(condition) This statement has the **full form**, or **abbreviated** (without else).

```
Full form: if (condition){ command_1; team_2; ... } else { command_1; team_2; ... }
Short form (without else): if (condition) { command 1; team 2; ... }
```

Block diagrams - see figure.





#### operation:

- symbol > greater than, x > y Boolean expression
- symbol < less than , x < y
- this == equals, x == y
- this >= greater than or equal to
- this <= less than or equal to
- this != NOT equal (<> is in other languages)
- these are all logical expressions. The logical expression takes the value:
- LIE (false) and TRUTH (true)

## Logical operators:

Logical operations connect (combine) various conditions with each other:

- Logical negation, also known as NOT or inversion. Indicated by the symbol "!" before the operand. Applies to one operand.
- Logical AND, also known as AND or conjunction. Identified by the symbol "&" between two operands, to which it applies. Example: x > 0 & x < 10
- Logical OR in Java, also known as OR, also known as disjunction. Denoted by the symbol "|" between two operands. Example:  $x < 0 \mid x > 10$  in this expression we get false when x = 2, 3, ... 9
- Exclusive or, XOR, strict disjunction. Denoted by a "^" symbol between two operands.
- Conditional OR, denoted by ||, tests the first operand (condition).
- Conditional AND, denoted &&, tests the first operand (condition). Conditional OR, conditional AND are useful when the right condition depends on the left one.

The result of comparisons and logical operators is always of type **boolean**, that is, it can be either TRUE or FALSE. There is no third!

## **Ternary operator**

"Ternary" means "triple". This is an alternative to the if-else conditional statement.

Earlier:

```
String securityAnswer;
    if(age>=18){
        securityAnswer="It's okay, come in!";
     }else{
        securityAnswer="This movie is not suitable for your age!";
     }
```

Now:





Example of design on several lines:

```
String securityAnswer=(age>=18)?
    "It's all right, come in!":
    "This film is not suitable for your age!";
```

A general recommendation might sound like this: if the condition is simple and easy to check, you can use the ternary operator without harm. This way you will reduce the amount of code and the number of if-else checks, of which there may already be a lot.

But if the condition is complex and multi-stage, it is better to use if-else.

For example, in this case it would be a bad idea to use the ternary operator:

```
String securityAnswer=(age>=18&&(hasTicket||hasCoupon)&&!hasChild)?"(
```

4

# Java Conditional and Logical Operators

## **Table for Comparison Operators**

Here is a table explaining comparison operators using two variables a = 10 and b = 20:

#### **Comparison Operator Example Result**

```
== a == b false
> a > b false
< a < b true
>= a >= b false
| a < b true
| a <= b true
| a != b true
```

## **Table for Logical Operators**

Here is a table for explaining logical operators "AND", "OR", and "XOR" using four variables a = 10, b = 20, c = 30, d = 40:

| <b>Logical Operator</b> |    | Example              | Result |
|-------------------------|----|----------------------|--------|
| & (AND)                 | (a | < b) & (c > d)       | false  |
| (OR)                    | (a | < b)   (c > d)       | true   |
| ^ (XOR)                 | (a | $< b) \land (c > d)$ | true   |





is true, but c > d is false, so the result is false.

- 2. (a < b) | (c > d) At least one of the conditions must be true for the result to be true. a < b is true, so the result is true.
- 3. (a < b) ^ (c > d) The result will be true if only one of the conditions is true. a < b is true, c > d is false, so the result is true.

## Tasks for Reinforcement:

- 1. Write a program that checks the user's age and displays a message indicating whether they can buy alcohol. Use the if-else conditional operator. (void ageVerification(int age))
- 2. Write a program that takes the temperature and suggests appropriate clothing for the weather. Use if-else and the ternary operator. (void weatherClothing(int temperature))
- 3. Write a program that checks the user's age and ticket possession. If the user has a ticket and is 18 or older, allow them to watch the movie. Use logical operators. (void movieTicketCheck(int age, boolean hasTicket))
- 4. Write a program that calculates discounts on items in the shopping cart. If the user has a coupon, add an additional 5% discount. Use if-else and logical operators. (void shoppingCartDiscount(double cartTotal, boolean hasCoupon))
- 5. Write a program that checks if you can go to a party. Conditions: you must be over 18 and either have an invitation or go with a friend who is already 21. Use logical operators. (void partyAccess(int age, boolean hasInvitation, boolean withFriend))

# Условный оператор

#### Оператор if()

Самым универсальный способ организовать выполнение какой-то части кода в зависимости от условия, является использования условного оператора: if(условие) {...} Этот оператор имеет **полную форму**, или **сокращенную** (без else).

```
Полная форма: if (условие){ команда_1; команда_2; ... } else { команда_1; команда_2; ... }
```

Сокращенная форма (без else): if (условие) { команда\_1; команда\_2; ... }

Блок-схемы - см. рисунок.

#### **Условие**

Условие в скобках оператора if () - это чаще всего операции сравнения:

• символ > больше, х > у - логическое выражение





- это != НЕ равно (<> это в других языках)
- всё это логические выражения. Логическое выражение принимает значение:
- ЛОЖЬ (false) или ПРАВДА (true)

### Логические операторы:

Логические операции соединяют(комбинируют) между собой различные условия:

- Логическое отрицание, оно же NOT или инверсия. Обозначается символом "!" перед операндом. Применяется к одному операнду.
- Логическое И, оно же AND или конъюнкция. Обозначается символом "&" между двумя операндами, к которым применяется. Пример: x > 0 & x < 10
- Логическое ИЛИ в Java, оно же ОR, оно же дизъюнкция. Обозначается символом "|" между двумя операндами. Пример: x < 0 | x > 10 в этом выражении получаем ложь, когда x = 2, 3, ... 9
- Исключающее или, XOR, строгая дизъюнкция. Обозначается символом "^" между двумя операндами.
- Условное ИЛИ, обозначаемое как ||, проверяет первый операнд (условие).
- Условное И, обозначаемое как &&, проверяет первый операнд (условие).
   Условное ИЛИ, условное И полезно тогда, когда правое условие зависит от левого.

Результат сравнения и логических операторов всегда имеет тип **boolean**, то есть может быть либо ИСТИНА (true), либо ЛОЖЬ (false). Третьего НЕ ДАНО!

# Тернарный оператор

"Тернарный" в переводе означает "тройной". Это альтернатива условному оператору if-else.

Раньше:

```
String securityAnswer;
    if(age>=18){
        securityAnswer="Все в порядке, проходите!";
    }else{
        securityAnswer="Этот фильм не подходит для вашего возраста!";
    }

Теперь:
```

String securityAnswer=(age>=18)?"Все в порядке, проходите!":"Этот фил





"Все в порядке, проходите!":

"Этот фильм не подходит для вашего возраста!";

Общая рекомендация может звучать так: если условие простое и легко проверяется, можно без ущерба использовать тернарный оператор. Таким образом ты сократишь объем кода и количество проверок if-else, которых там и без того может быть немало.

А вот если условие является сложным и многоступенчатым — лучше использовать if-else.

Например, в этом случае использовать тернарный оператор будет плохой идеей:

String securityAnswer=(age>=18&&(hasTicket||hasCoupon)&&!hasChild)?"[

4

# Дополнительный материал:

# Java Условные и Логические Операторы

## Таблица Операторов Сравнения

Вот таблица, объясняющая операторы сравнения с использованием двух переменных a = 10 и b=20:

#### **Comparison Operator Example Result**

| == | a == b | false | Переменные а и b не равны между собой |
|----|--------|-------|---------------------------------------|
| >  | a > b  | false | а меньше b                            |
| <  | a < b  | true  | а меньше b                            |
| >= | a >= b | false | а меньше b                            |
| <= | a <= b | true  | а меньше или равно b                  |
| != | a != b | true  | а не равно b                          |

#### Таблица Логических Операторов

Вот таблица для объяснения логических операторов "И", "ИЛИ" и "ИСКЛЮЧАЮЩЕЕ ИЛИ" с использованием четырёх переменных a=10,b=20, c=30, d=40:





| ,       | d)                  | с>d ложно                           |
|---------|---------------------|-------------------------------------|
| l (OR)  | (a < b)   (c > true | Хотя бы одно из условий должно быть |
| 1 (01)  | d)                  | ИСТИННЫМ                            |
| ^ (XOR) | (a < b) ^ (c > true | Только одно из условий должно быть  |
| (XON)   | d)                  | истинным                            |

#### **Explanations:**

- 1. (a < b) & (c > d) Both conditions must be true for the result to be true. a < b is true, but c > d is false, so the result is false.
- 2. (a < b) | (c > d) At least one of the conditions must be true for the result to be true. a < b is true, so the result is true.
- 3. (a < b) ^ (c > d) The result will be true if only one of the conditions is true. a < b is true, c > d is false, so the result is true.

## Задачи для закрепления:

- 1. Напишите программу, которая проверяет возраст пользователя и выводит сообщение, может ли он купить алкоголь. Используйте условный оператор if-else. (void ageVerification(int age))
- 2. Напишите программу, которая принимает температуру и выводит подходящую одежду для погоды. Используйте if-else и тернарный оператор. (void weatherClothing(int temperature))
- 3. Напишите программу, которая проверяла бы возраст, наличие билета у пользователя. Если у пользователя есть билет, и пользователю 18 или больше лет, то пускайте его на фильм. Используйте логические операторы. (void movieTicketCheck(int age, boolean hasTicket))
- 4. Напишите программу, которая рассчитывает скидку на товары в корзине. Если у пользователя есть купон, добавьте дополнительную скидку 5%. Используйте if-else и логические операторы. (void shoppingCartDiscount(double cartTotal, boolean hasCoupon))
- 5. \*Напишите программу, которая проверяет, можно ли вам пойти на вечеринку. Условия: вы должны быть старше 18 лет и у вас должно быть либо приглашение, либо вы идете с другом, который уже 21 год. Используйте логические операторы. (void partyAccess(int age, boolean hasInvitation, boolean withFriend))

#### 

**Задача 1.** Задать в программе три стороны треугольника. Проверить выполнимость неравенства треугольника - любая из сторон должна быть меньше суммы двух других. Сообщить результат - существует или нет треугольник с заданными сторонами.





дополнительная скидка 5%. Вычислите итоговую стоимость покупки и величину полученной скидки.

**Задача 3(\*).** Написать программу, определяющую, является ли год високосным. Для того чтобы год был високосным, он должен: делится без остатка на 4 и, либо не должен делится без остатка на 100, либо если он делится на 100, то он должен делится также на 400. Проверить на годах 1700, 1800, 1900 - не високосные, а 2000 високосный.

#### ∨ Code

hw/HwSolution/src/HwSolution.java

```
/*
```

- 1. Разбор домашнего задания:
- 2. Метод вычисляющий площадь круга;
- 3. Метод вычисляющий площадь квадрата;

#### Дополнительные задачи:

- 4. Найдите в Интернете данные по температуре в вашем городе за проше!
- 5. Пусть цена товара А обычно составляет 1000 Euro, и товара В состав
- 6. Какие числа могут получаться в качестве остатка от деления какого-
- 7. Создайте три переменные num1, num2 и num3 типа double и присвойте

```
8. Объявите 10 переменных типа int со значениями 0,1, 2, 3, 4, 5, 6,
*/
public class HwSolution {
    public static void main(String[] args) {
        System.out.println("1. Метод вычисляющий площадь круга");
        circleArea(5.14);
        double radius = 185.5;
        circleArea(radius);
        System.out.println("2. Метод вычисляющий площадь квадрата");
        squareArea(11);
        System.out.println("3. Метод вычисляющий площадь прямоугольны
        rectangleArea(15, 30);
        rectangleArea(150, 3000);
        int siteA = 120;
        int siteB = 1250;
        rectangleArea(siteA, siteB);
        squareArea(siteA);
        circleArea(siteB);
        System.out.println("Доступные стороны a = " + siteA + " b = '
```





```
System.out.println("5. Пусть цена товара А обычно составляет
    calculateTotalPriceWithDiscount(1000, 500);
    System.out.println("6. Какие числа могут получаться в качесты
    calculateRemainder(10, 2);
    calculateRemainder(9, 2);
    calculateRemainder(8, 4);
    System.out.println("7. Создайте три переменные num1, num2 и r
    average(2, 3, 4);
    System.out.println("8. Объявите 10 переменных типа int со зна
    averageFor10();
}
// 1. Метод вычисляющий площадь круга
public static void circleArea(double radius) {
    double pi = 3.14;
    // формула = pi * (radius^2)
    double resultAsDouble = pi * (radius * radius);
    System.out.println("Площадь круга = " + resultAsDouble + " к
}
// 2. Метод вычисляющий площадь квадрата
public static void squareArea(int site) {
    int result = site * site;
    System.out.println("Площадь квадрата = " + result + " кв. см.
}
// 3. Метод вычисляющий площадь прямоугольника
public static void rectangleArea(int siteA, int siteB) {
    int result = siteA * siteB;
    System.out.println("Площадь прямоугольника = " + result + " +
}
// 4. Найдите в Интернете данные по температуре в вашем городе за
// Вычислите среднюю температуру за неделю и выведете ее на печат
public static void calculateAverageTemperature(double temperature
    double sum = temperature1 + temperature2 + temperature3 + tem
    double average = sum / 7;
    System.out.println("среднюю температуру за неделю " + average
}
// 5. Пусть цена товара А обычно составляет 1000 Euro, и товара Е
```





```
double discount = 0.1;
          double totalBeforeDiscount = priceA + priceB;
          double discountAmount = totalBeforeDiscount * discount;
          double totalAfterDiscount = totalBeforeDiscount - discountAm
          System.out.println("Стоимость суммы A+B со скидкой: " + total
          System.out.println("Сумма скидки: " + discountAmount);
      }
     // 6. Какие числа могут получаться в качестве остатка от деления
      public static void calculateRemainder(int dividend, int divisor)
          int remainder = dividend % divisor;
          System.out.println("Остаток от деления " + dividend + " на "
      }
     // 7. Создайте три переменные num1, num2 и num3 типа double и при
      public static void average(int num1, int num2, int num3) {
          double average = (num1 + num2 + num3) / 3;
          System.out.println("Среднее значение: " + average);
      }
     // 8. Объявите 10 переменных типа int со значениями 0,1, 2, 3, 4,
      public static void averageFor10() {
          int num1 = 1;
         int num2 = 2;
         int num3 = 3;
         int num4 = 4;
          int num5 = 5;
          int num6 = 6;
          int num7 = 7;
          int num8 = 8;
          int num9 = 9;
          int num10 = 10;
          double average = (num1 + num2 + num3 + num4 + num5 + num6 + r
          System.out.println("Среднее арифметическое: " + average);
      }
 }
code/Conditional_operator/src/Main.java
 /**
   * @author Andrej Reutow
   * created on 11.09.2023
```





```
// boolean может хранить только true, talse
//
          boolean varTrue = true;
          boolean varFalse = false;
//
//
//
          if (varTrue) {
//
              System.out.println("If оператор вариант 1");
              System.out.println("Условие (вариант 1) - true");
//
//
          }
//
//
          if (varTrue) {
//
              System.out.println("If оператор вариант 2");
//
              System.out.println("Условие (вариант 2) - true");
          } else {
//
              System.out.println("Условие - true, мы в блоке else");
//
//
          }
//
//
          if (varFalse) {
//
              System.out.println("If оператор");
//
              System.out.println("Условие - false");
//
//
          } else {
              System.out.println("Условие - false, мы в блоке else");
//
//
          }
        task1();
    }
    /*
    Условие в скобках оператора if () - это чаще всего операции сравн
    символ > больше, х > у - логическое выражение
    символ < меньше, х < у
    это == равно, x == y
    это >= больше или равно
    это <= меньше или равно
    это != НЕ равно (<> - это в других языках)
    всё это логические выражения. Логическое выражение принимает знач
    ЛОЖЬ (false) или ПРАВДА (true)
    */
    public static void task1() {
//
          //if - ECЛИ, ELSE - иначе
        System.out.println();
        compareGreaterThen(10, 10);
        compareGreaterThen(11, 10);
        compareGreaterThen(9, 10);
```





```
compareLessIhen(11, 10);
    compareLessThen(9, 10);
    System.out.println();
    compare(10, 10);
    compare(11, 10);
    compare(9, 10);
    System.out.println();
    compareGreaterThanOrEqualTo(10, 10); // true
    compareGreaterThanOrEqualTo(11, 10); // true
    compareGreaterThanOrEqualTo(9, 10); // false
    System.out.println();
    compareLessThanOrEqualTo(10, 10); // true
    compareLessThanOrEqualTo(11, 10); // false
    compareLessThanOrEqualTo(9, 10); // true
    System.out.println();
    compareNotEqualTo(10, 10); // false
    compareNotEqualTo(11, 10); // true
    compareNotEqualTo(9, 10); // true
}
// символ > больше, x > y - логическое выражение
public static void compareGreaterThen(int numberA, int numberB) +
    if (numberA > numberB) {
        System.out.println(numberA + " больше чем " + numberB);
    } else {
        System.out.println(numberA + " не больше чем " + numberB)
    }
}
// символ < меньше, x < y
public static void compareLessThen(int numberA, int numberB) {
    if (numberA < numberB) {</pre>
        System.out.println(numberA + " меньше чем " + numberB);
        System.out.println(numberA + " не меньше чем " + numberB)
    }
}
// это == равно,х == у
public static void compare(int numberA, int numberB) {
    if (numberA == numberB) {
```





```
}
      // >= - больше или равно
      // numberA = 10, numberB = 10 -- true
      // numberA = 11, numberB = 10 -- true
      // numberA = 9, numberB = 10 -- false
      public static void compareGreaterThanOrEqualTo(int numberA, int r
          if (numberA >= numberB) {
              System.out.println(numberA + " больше либо равно, чем " -
          } else {
              System.out.println(numberA + " не больше либо не равно, ч
          }
      }
      // <= - меньше или равно
     // numberA = 10, numberB = 10 -- true
      // numberA = 11, numberB = 10 -- false
      // numberA = 9, numberB = 10 -- true
      public static void compareLessThanOrEqualTo(int numberA, int numberA)
          if (numberA <= numberB) {</pre>
              System.out.println(numberA + " меньше либо равно, чем " +
          } else {
              System.out.println(numberA + " не меньше либо не равно, ч
          }
      }
     // != - не равно
      // numberA = 10, numberB = 10 -- compareResult = false
     // numberA = 11, numberB = 10 -- compareResult = true
      // numberA = 9, numberB = 10 -- compareResult = true
      public static void compareNotEqualTo(int numberA, int numberB) {
          boolean compareResult = numberA != numberB;
          if (compareResult) {
              System.out.println(numberA + " не равно " + numberB);
          } else {
              System.out.println(numberA + " равно " + numberB);
          }
          System.out.println("Результат сравнения: " + compareResult);
          // System.out.println("Результат сравнения: " + (numberA !=
      }
 }
code/Conditional_operator/src/LogicalOperators.java
```





```
*/
public class LogicalOperators {
    public static void main(String[] args) {
        checkKino(true, 30); // true
        checkKino(false, 30); // false
        checkKino(true, 11); // false
        checkKino(false, 11); // false
        System.out.println();
        checkKinoWithCoupon(true, true); // true
        checkKinoWithCoupon(true, false); // true
        checkKinoWithCoupon(false, false);// false
        checkKinoWithCoupon(false, true); // true
        // boolean isAllowed = (hasTicket | hasCoupon) & age >= 18;
        checkKinoWithCouponAnAge(true, true, 18); // true
        checkKinoWithCouponAnAge(false, true, 20); // true
        checkKinoWithCouponAnAge(true, false, 21); // true
        checkKinoWithCouponAnAge(true, true, 17); // false
        checkKinoWithCouponAnAge(false, false, 30); // false
        checkKinoWithCouponAnAge(false, true, 30); // true
    }
/*
Логические операторы:
Логические операции соединяют(комбинируют) между собой различные усло
Логическое отрицание, оно же NOT или инверсия. Обозначается символом
Логическое И, оно же AND или конъюнкция. Обозначается символом "&" мє
Логическое ИЛИ в Java, оно же - ОR, оно же - дизъюнкция. Обозначается
Исключающее или, XOR, строгая дизъюнкция. Обозначается символом "^" м
Условное ИЛИ, обозначаемое как ||, проверяет первый операнд (условие)
Условное И, обозначаемое как &&, проверяет первый операнд (условие).
Результат сравнения и логических операторов всегда имеет тип boolean,
*/
    public static void checkKino(boolean hasTicket, int age) {
        boolean isAllowed = hasTicket & age >= 18;
        if (isAllowed) {
            System.out.println("Имеет право посмотреть фильм");
        } else {
            System.out.println("Не имеет право посмотреть фильм");
```





```
boolean isAllowed = haslicket | hasCoupon;
          if (isAllowed) {
              System.out.println("Имеет право посмотреть фильм");
          } else {
              System.out.println("Не имеет право посмотреть фильм");
          }
      }
      public static void checkKinoWithCouponAnAge(boolean hasTicket, bo)
          boolean isAllowed = (hasTicket | hasCoupon) & age >= 18;
 //
            boolean isAllowed = (hasTicket || hasCoupon) & age >= 18;
          // boolean isAllowed = age >= 18 & (hasTicket | hasCoupon);
          // false | false | true | false | false // true
          // false || false || true || (false) || (false) // true
          if (isAllowed) {
              System.out.println("Имеет право посмотреть фильм");
          } else {
              System.out.println("He имеет право посмотреть фильм");
          }
      }
 }
code/Conditional_operator/src/TernaryOperator.java
 /**
  * @author Andrej Reutow
  * created on 11.09.2023
  */
 public class TernaryOperator {
      public static void main(String[] args) {
 //
            String message = checkAge(20);
 //
            System.out.println(message);
          int age = 17;
          String messageV2 =
                  age > 17 ?
                          "Все в порядке, проходите!" :
                          "Этот фильм не подходит для вашего возраста!"
          System.out.println(messageV2);
      }
```





```
1† (age > 1/) {
    message = "Все в порядке, проходите!";
} else {
    message = "Этот фильм не подходит для вашего возраста!";
}

return message;
}
```

Legacy lesson view

