

## Лаб: Условни конструкции

Задачи за упражнение в клас и за домашно към курса "[Основи на програмирането](#)" @ СофтУни.

Тествайте решенията си в Judge системата: <https://judge.softuni.bg/Contests/Compete/Index/1012>

### 1. Проверка за отлична оценка

Напише конзолна програма, която чете оценка (реално число), въведена от потребителя, и отпечатва "Excellent!" ако оценката е 5.50 или по-висока.

вход	изход
6	Excellent!

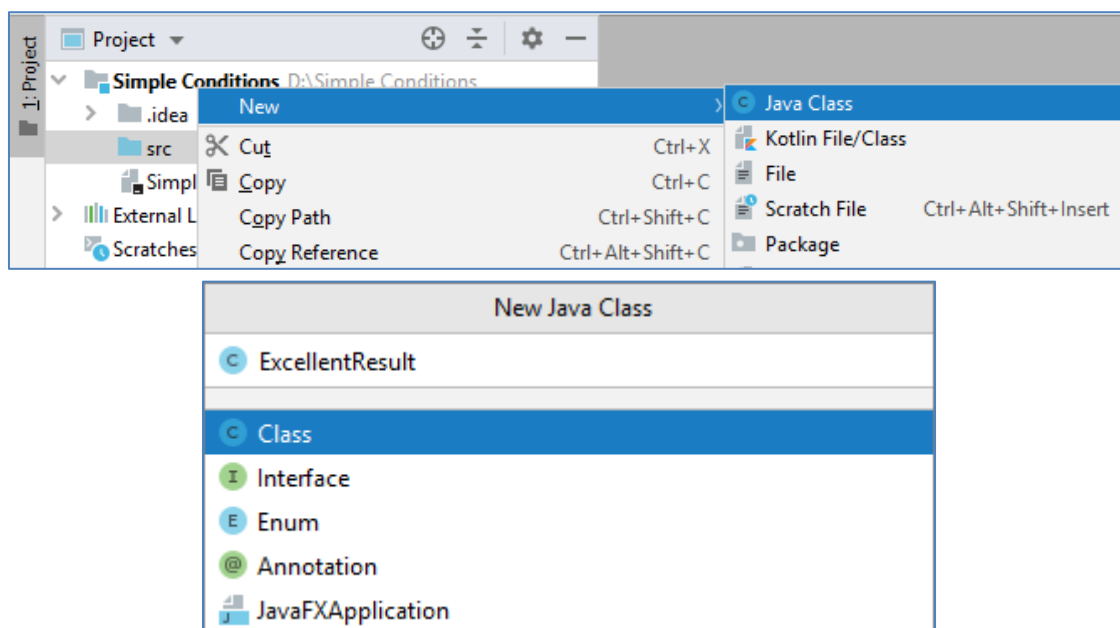
вход	изход
5	(няма изход)

вход	изход
5.50	Excellent!

вход	изход
5.49	(няма изход)

#### Насоки:

1. Създайте **нов клас** в съществуващия IntelliJ проект. Кликнете с десен бутон на мишката върху папката "src". Изберете [New] → [Java Class]:



Вече имате проект с един клас в него. Остава да напишете кода за решаване на задачата.

2. Създайте **main метод** като отидете в класа "ExcellentResult" (между квадратните скоби) и напишете:

```
public class ExcellentResult {  
    public static void main(String[] args) {  
  
    }  
}
```

3. Отидете в тялото на метода `main(String[] args)` (между къдравите скоби). Създайте **Scanner** обект, с който да четете от конзолата и прочетете едно **реално число** - оценката:

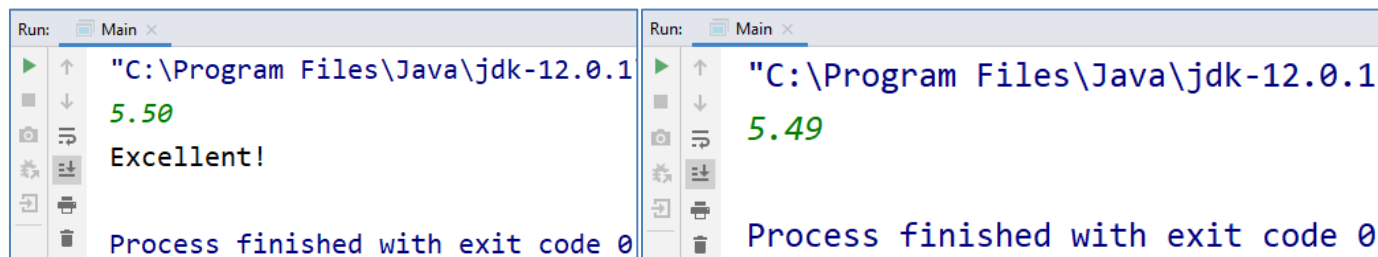
```
public class ExcellentResult {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        double grade = Double.parseDouble(scan.nextLine());
    }
}
```

4. Направете проверка за стойността на оценката. Ако тя е по-голяма или равна на 5.50, отпечатайте изхода по условие:

```
public class Main {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        double grade = Double.parseDouble(scan.nextLine());

        if (grade >= 5.50) {
            System.out.println("Excellent!");
        }
    }
}
```

5. **Стартирайте** програмата с [Ctrl+Shift+F10] и я **тествайте** с различни входни стойности:



## 2. Намиране на по-голямото число

Да се напише програма, която чете **две цели числа**, въведени от потребителя, и отпечатва **по-голямото от двете**.

### Примерен вход и изход

вход	изход
5 3	5

вход	изход
3 5	5

вход	изход
10 10	10

вход	изход
-5 5	5

### Насоки:

1. Прочетете 2 цели числа от конзолата:

```
public class GreaterNumber {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        int num1 = Integer.parseInt(scan.nextLine());
        int num2 = Integer.parseInt(scan.nextLine());
    }
}
```

- Сравнете дали първото число **num1** е по-голямо от второто **num2**. Отпечатайте по-голямото число.

```
if (num1 > num2) {
    System.out.println(num1);
} else {
    System.out.println(num2);
}
```

### 3. Четно или нечетно

Да се напише програма, която чете **цяло число**, въведено от потребителя, и печата дали е **четно** или **нечетно**.

#### Примерен вход и изход

вход	изход
2	even

вход	изход
3	odd

вход	изход
25	odd

вход	изход
1024	even

#### Насоки:

- Първо добавете **нов Java клас** към съществуващия проект
- Създайте **Scanner** обект и прочетете едно цяло число от конзолата:

```
public class OddOrEven {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        int number = Integer.parseInt(scan.nextLine());
    }
}
```

- Проверете дали числото е четно като извършите деление на 2 с остатък и проверете дали има остатък от делението. Отпечатайте изхода по условие – текста **"even"**:

```

public class OddOrEven {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        int number = Integer.parseInt(scan.nextLine());

        if (number % 2 == 0) {
            System.out.println("even");
        }
    }
}

```

4. В противен случай отпечатайте "odd":

```

public class OddOrEven {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        int number = Integer.parseInt(scan.nextLine());

        if (number % 2 == 0) {
            System.out.println("even");
        } else {
            System.out.println("odd");
        }
    }
}

```

## 4. Изписване на число до 9 с думи

Да се напише програма, която чете **цяло число в диапазона [1...9]**, въведено от потребителя, и го **изписва с думи** на английски език. Ако числото е извън диапазона, изписва "number too big".

### Примерен вход и изход

вход	изход
5	five

вход	изход
1	one

вход	изход
9	nine

вход	изход
10	number too big

## 5. Число от 100 до 200

Да се напише програма, която **чете цяло число**, въведено от потребителя, и проверява дали е **под 100, между 100 и 200** или **над 200**. Да се отпечатаат съответно съобщения като в примерите по-долу:

### Примерен вход и изход

вход	изход
95	Less than 100

вход	изход
120	Between 100 and 200

вход	изход
210	Greater than 200

## 6. Познай паролата

Да се напише програма, която **чете парола** (един ред с произволен текст), въведена от потребителя, и проверява дали въведеното **съвпада** с фразата "s3cr3t!P@ssw0rd". При съвпадение да се изведе "Welcome". При несъвпадение да се изведе "Wrong password!".

## Примерен вход и изход

вход	изход
qwerty	Wrong password!

вход	изход
s3cr3t!P@ssw0rd	Welcome

вход	изход
s3cr3t!p@ss	Wrong password!

## 7. Лица на фигури

Да се напише програма, в която потребителят **въвежда вида и размерите на геометрична фигура** и пресмята лицето ѝ. Фигурите са четири вида: квадрат (**square**), правоъгълник (**rectangle**), кръг (**circle**) и триъгълник (**triangle**). На първия ред на входа се чете вида на фигурата (**square, rectangle, circle** или **triangle**). Ако фигурата е **квадрат**, на следващия ред се чете едно число - дължина на страната му. Ако фигурата е **правоъгълник**, на следващите два реда четат две числа - дължините на страните му. Ако фигурата е **кръг**, на следващия ред се чете едно число - радиусът на кръга. Ако фигурата е **триъгълник**, на следващите два реда четат две числа - дължината на страната му и дължината на височината към нея. Резултатът да се закръгли до **3 цифри след десетичната точка**.

## Примерен вход и изход

вход	изход
square 5	25.000

вход	изход
rectangle 7 2.5	17.500

вход	изход
circle 6	113.097

вход	изход
triangle 4.5 20	45.000

## 8. Ден от седмицата

Напишете програма, която чете **цяло число**, въведено от потребителя, и отпечатва **ден от седмицата** (на английски език), в граници [1...7] или отпечатва **"Error"** в случай, че въведеното число е **невалидно**.

## Примерен вход и изход

Вход	Изход
1	Monday
2	Tuesday
3	Wednesday
4	Thursday
5	Friday
6	Saturday
7	Sunday
-1	Error

\*Подсказка: Използвайте условната конструкция [switch-case](#).

## 9. Клас животно

Напишете програма, която отпечатва класа на животното според неговото име, въведено от потребителя.

- dog -> mammal
- crocodile, tortoise, snake -> reptile
- others -> unknown

## Примерен вход и изход

Вход	Изход
dog	mammal
snake	reptile
cat	unknown

\*Подсказка: Използвайте условната конструкция **switch-case**.

## Примерна изпитна задача

### 10. Магазин за детски играчки

Петя има магазин за детски играчки. Тя получава голяма поръчка, която трябва да изпълни. С парите, които ще спечели иска да отиде на екскурзия. Да се напише програма, която пресмята печалбата от поръчката.

Цени на играчките:

- Пъзел - 2.60 лв.
- Говореща кукла - 3 лв.
- Плюшено мече - 4.10 лв.
- Миньон - 8.20 лв.
- Камионче - 2 лв.

Ако поръчаните играчки са **50 или повече** магазинът прави **отстъпка 25% от общата цена**. От спечелените пари Петя трябва да даде **10% за наема** на магазина. Да се пресметне дали парите ще ѝ стигнат да отиде на екскурзия.

### Вход

От конзолата се четат **6** реда:

1. Цена на екскурзията - реално число в интервала [1.00 ... 10000.00]
2. Брой пъзели - цяло число в интервала [0... 1000]
3. Брой говорещи кукли - цяло число в интервала [0 ... 1000]
4. Брой плюшени мечета - цяло число в интервала [0 ... 1000]
5. Брой миньони - цяло число в интервала [0 ... 1000]
6. Брой камиончета - цяло число в интервала [0 ... 1000]

### Изход

На конзолата се отпечатва:

- Ако парите са достатъчни се отпечатва:
  - "Yes! {оставащите пари} lv left."
- Ако парите НЕ са достатъчни се отпечатва:
  - "Not enough money! {недостигащите пари} lv needed."

Резултатът трябва да се форматира до втория знак след десетичната запетая.

## Примерен вход и изход

Вход	Изход	Обяснения
------	-------	-----------



40.8 20 25 30 50 10	Yes! 418.20 lv left.	<p>Сума: <math>20 * 2.60 + 25 * 3 + 30 * 4.10 + 50 * 8.20 + 10 * 2 = 680</math> лв.</p> <p>Брой на играчките: <math>20 + 25 + 30 + 50 + 10 = 135</math></p> <p><math>135 &gt; 50 \Rightarrow 25\%</math> отстъпка; <math>25\%</math> от 680 = <b>170</b> лв. отстъпка</p> <p>Крайна цена: <math>680 - 170 = 510</math> лв.</p> <p>Наем: <math>10\%</math> от 510 лв. = <b>51</b> лв.</p> <p>Печалба: <math>510 - 51 = 459</math> лв.</p> <p><math>459 &gt; 40.8 \Rightarrow 459 - 40.8 = 418.20</math> лв. остават</p>
Вход	Изход	Обяснения
320 8 2 5 5 1	Not enough money! 238.73 lv needed.	<p>Сума: <b>90.3</b> лв.</p> <p>Брой на играчките: <b>21</b></p> <p><math>21 &lt; 50 \Rightarrow</math> няма отстъпка</p> <p>Наем: <math>10\%</math> от 90.3 = <b>9.03</b> лв.</p> <p>Печалба: <math>90.3 - 9.03 = 81.27</math> лв.</p> <p><math>81.27 &lt; 320 \Rightarrow 320 - 81.27 = 238.73</math> лв. не достигат</p>