



**Преговор**

1. Какъв е типът на променливата:

```
... letter = 'a';
```

String

int

char

double

1. Какъв е типът на променливата:

```
... letter = 'a';
```

String

int

char

double

2. Какъв е типът на променливата:

```
... number = "1000";
```

String

int

char

double

2. Какъв е типът на променливата:

```
... number = "1000";
```

String

int

char

double

3. Как се нарича долепването на два текста (низа)?



Събиране

Конкатенация

Кулминация

Съединяване

3. Как се нарича долепването на два текста (низа)?



Събиране

Конкатенация

Кулминация

Съединяване

4. Какво ще се отпечата на конзолата, ако изпълним следната команда:

```
System.out.println(10 % 3);
```

10

1

0

3



4. Какво ще се отпечата на конзолата, ако изпълним следната команда:

```
System.out.println(10 % 3);
```

10

1

0

3

5. Каква стойност държи променливата **result**:

```
int a = 5;  
int b = 2;  
double result = a / b;
```

2.5

7

2.0

1

5. Каква стойност държи променливата **result**:

```
int a = 5;  
int b = 2;  
double result = a / b;
```

2.5

7

2.0

1

6. Какъв би бил резултатът, ако се опитаме да изпълним следната команда:

```
System.out.println(1 + 1 + "4" + 2 + 1);
```

Compile  
time error

9

243

2421

6. Какъв би бил резултатът, ако се опитаме да изпълним следната команда:

```
System.out.println(1 + 1 + "4" + 2 + 1);
```

Compile  
time error

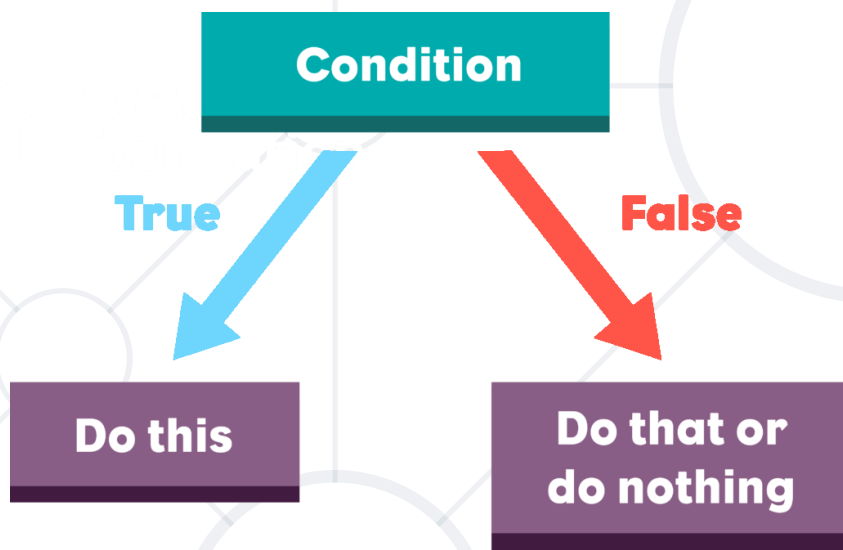
9

243

2421

# Проверки

Логически изрази и проверки. Условна конструкция if-else



СофтУни

Преподавателски екип



Software  
University



SoftUni  
Foundation



Software University

<http://softuni.bg>

# Имате въпроси?

[sli.do](https://sli.do)

**#pb-nov**



# Логически изрази и проверки

Оператори за сравнение



# Оператори за сравнение



SoftUni  
Foundation

Оператор	Означение	Работи за
Равенство	==	числа, дати, други сравними типове
Различно	!=	
По-голямо	>	
По-голямо или равно	>=	
По-малко	<	
По-малко или равно	<=	



- В програмирането можем да сравняваме стойности
  - Резултатът от логическите изрази е **true** или **false**

```
int a = 5;  
int b = 10;  
System.out.println(a < b);           // true  
System.out.println(a > 0);           // true  
System.out.println(a > 100);         // false  
System.out.println(a < a);           // false  
System.out.println(a <= 5);          // true  
System.out.println(b == 2 * a);      // true
```



# Сравняване на стойности (2)

- Сравняване на текст чрез `==` по адрес в паметта

```
String a = "Example";  
String b = a;  
System.out.println(a == b);    // true
```

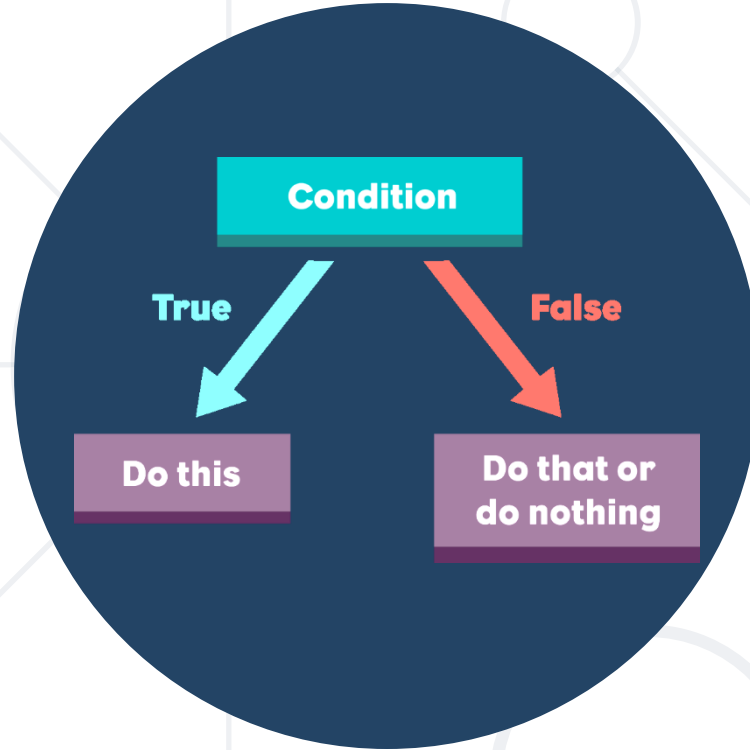
```
Scanner scanner = new Scanner(System.in);  
String a = scanner.nextLine();  
String b = scanner.nextLine();  
System.out.println(a == b);    // false
```

Въвеждане на  
еднаква стойност

- Променливи от тип **String** сравняваме чрез метода **equals**
  - Сравняване на текст чрез **equals** по стойност:

```
Scanner scanner = new Scanner(System.in);  
String a = scanner.nextLine();  
String b = scanner.nextLine();  
System.out.println(a.equals(b)); // true
```

Въвеждане на  
еднаква стойност




# Условни конструкции

## Прости проверки

# Прости проверки

- Често проверяваме условия и извършваме действия според резултата



Условие  
(булев израз)

```
if (...) {  
    // код за изпълнение  
}
```

Код за изпълнение при  
вярност на условието

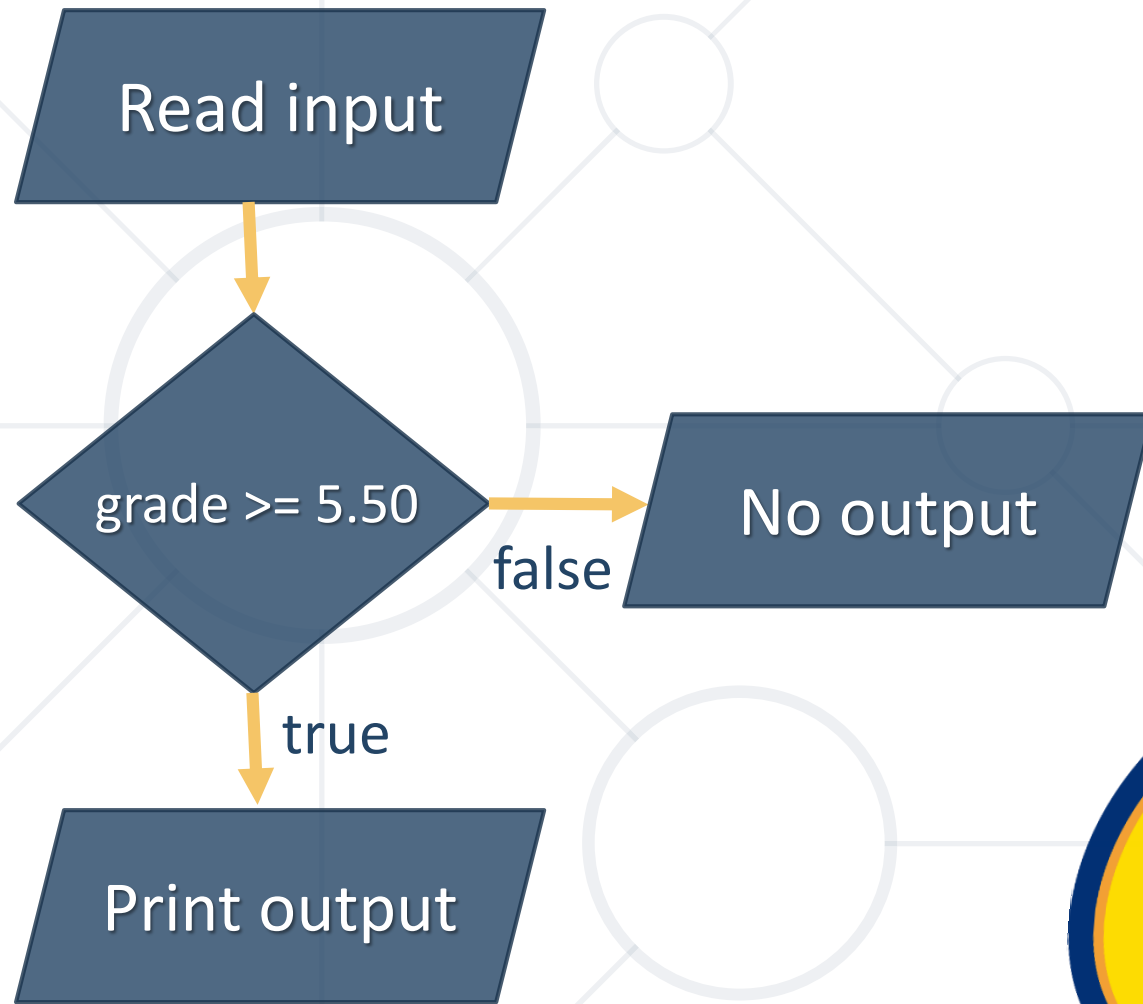
- Резултатът е **true** или **false**

- Напишете **програма**, която:
  - **Чете** оценка (**число**), въведена от потребителя
  - **Проверява** дали е отлична
  - **Отпечатва на конзолата** "Excellent", ако оценката е по-голяма или равна на 5.50

- Пример:



236



6



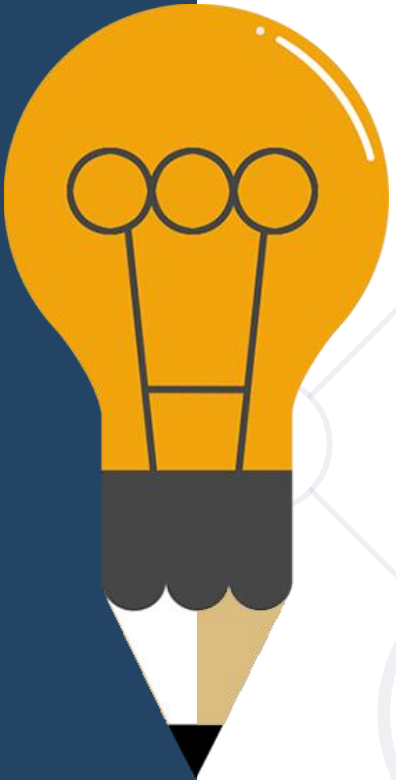


# Прости проверки – if-else

- При **невярност** (false) на условието, можем да изпълним други действия – чрез **else** конструкция

```
if (...) {  
    // код за изпълнение  
} else {  
    // код за изпълнение  
}
```

Код за изпълнение  
при невярност на  
условието



- Къдравите скоби { } въвеждат блок (група команди)

```
String color = "red";  
if (color.equals("red"))  
    System.out.println("tomato");  
else  
    System.out.println("banana");  
System.out.println("bye");
```

```
"C:\Program Files\Java\jdk-11.0.2\  
tomato  
bye  
  
Process finished with exit code 0
```

Изпълнява се винаги – не е част от if/else конструкцията

- Ако **включим скоби**, се изпълнява съответния блок

```
String color = "red";  
if (color.equals("red")) {  
    System.out.println("tomato");  
} else {  
    System.out.println("banana");  
    System.out.println("bye");  
}
```

Изпълняват се редовете  
в съответния блок

```
"C:\Program Files\Java\jdk-11.0.2  
tomato  
  
Process finished with exit code 0
```

# По-голямото число – условие

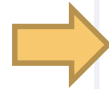
- Напишете програма, която:
  - Чете две **цели** числа
  - Извежда **"Greater number: "**
  - Отпечатва на конзолата **по-голямото** от тях
- Пример:

5  
8



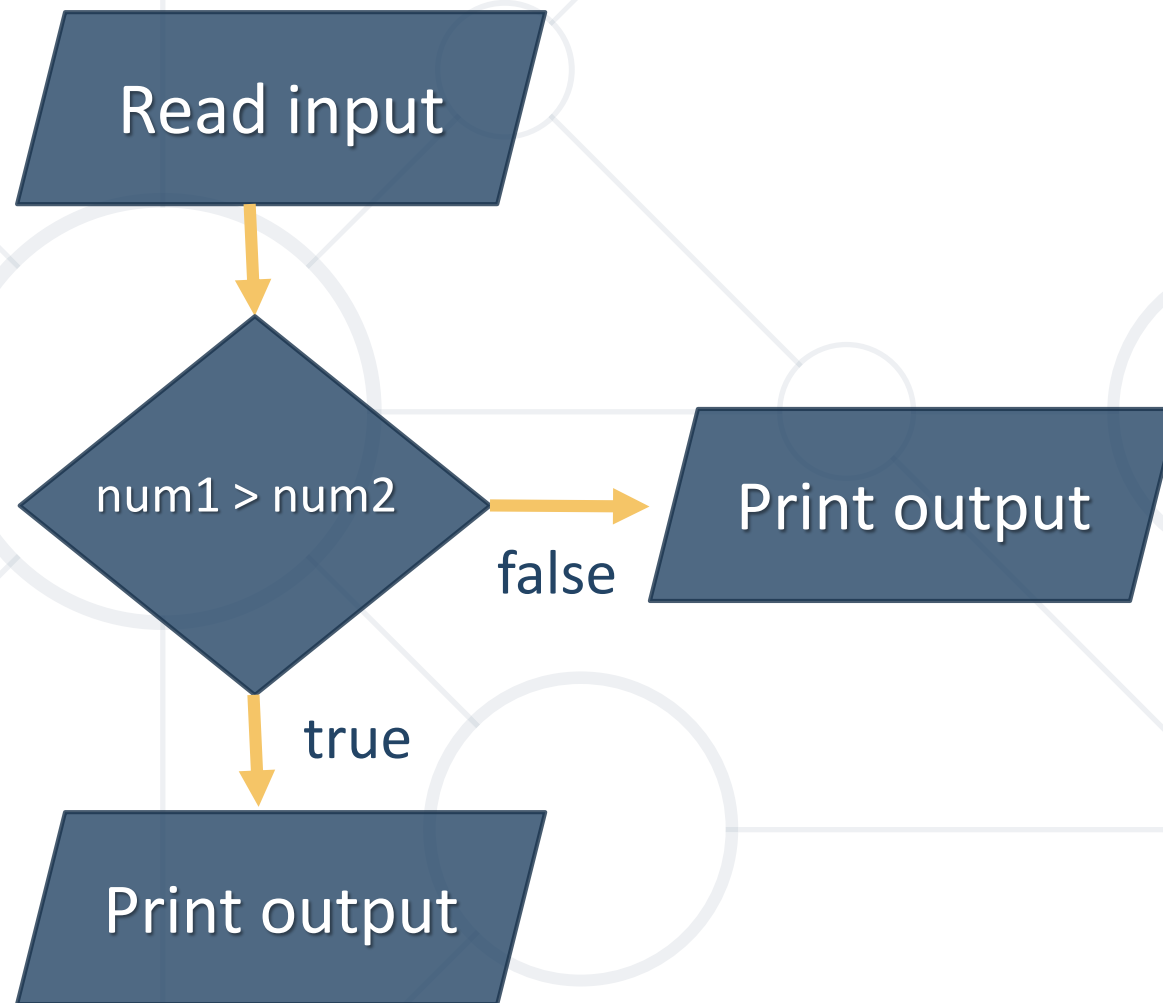
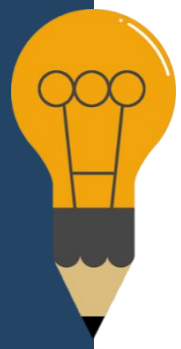
8

7  
3



7





Тестване на решението: <https://judge.softuni.bg/Contests/Compete/Index/1012#1>

# Четно или нечетно число – условие

- Напишете програма, която:
  - Проверява дали едно число е **четно** или **нечетно**
  - Ако е четно отпечатва на конзолата **"even"**
  - Ако е нечетно отпечатва на конзолата **"odd"**
- Пример:

4 → even

7 → odd

4 7

# Четно или нечетно – решение

```
Scanner scanner = new Scanner(System.in);  
int num = Integer.parseInt(scanner.nextLine());  
if (num % 2 == 0) {  
    System.out.println("even");  
} else {  
    System.out.println("odd");  
}
```




**Серии от проверки**  
**По-сложни условни конструкции**



# Сerii от проверки

- Конструкцията **if/else-if/else...** е серия от проверки




```
if (...)  
    // код за изпълнение  
else if (...)  
    // код за изпълнение  
else if (...)  
    // код
```

**TRUE** OR **FALSE?**

- При истинност на едно условие, **не се продължава** към проверяване на следващите условия

# Серия от проверки - пример

- Програмата проверява първото условие, установява, че е вярно и приключва



```
int a = 7;  
if (a > 4)  
    System.out.println("Bigger than 4");  
else if (a > 5)  
    System.out.println("Bigger than 5");  
else  
    System.out.println("Equal to 7");
```

Извежда на  
конзолата само  
"Bigger than 4"

# Число от 1 до 9 с текст - условие

- Напишете програма, която:
  - Чете **цяло число**, въведено от потребителя
  - Проверява неговата стойност [1, 9]
  - Ако числото е по-голямо от 9 отпечатва "number too big"
  - Отпечатва стойността **с текст**
- Пример:

7 → seven

10 → number too big

# Число от 1 до 9 с текст - решение

```
Scanner scanner = new Scanner(System.in);
int num = Integer.parseInt(scanner.nextLine());
if (num == 1)
    System.out.println("one");
else if (num == 2)
    System.out.println("two");
else if (num == 3)
    System.out.println("three");
// TODO: Add more conditional statements
else
    System.out.println("number too big");
```

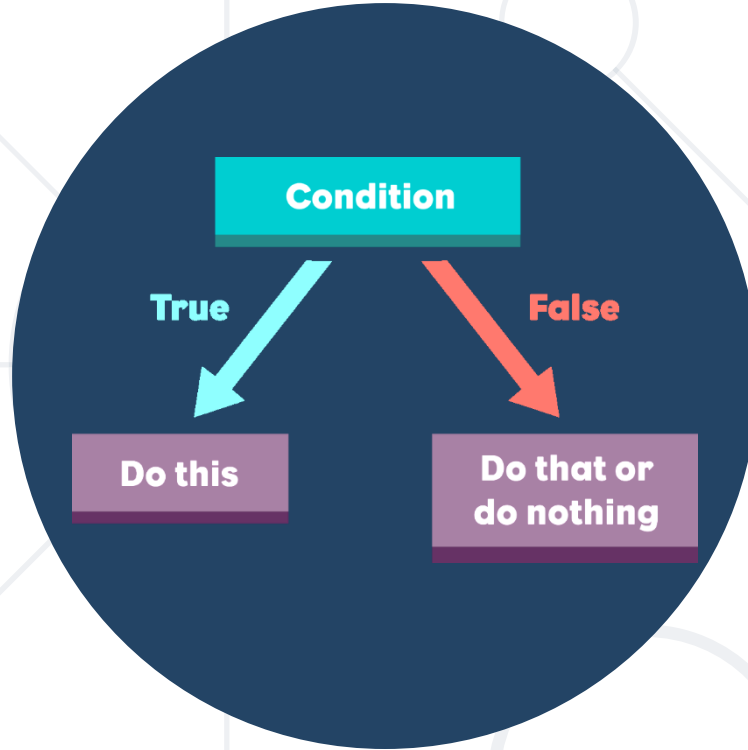


**Живот на променлива**

**Диапазон на използване**

- Обхват, в който може да бъде използвана
  - Пример: Променливата **salary** съществува **само** в блока от код на **if**-конструкцията

```
String currentDay = "Monday";  
if (currentDay.equals("Monday")) {  
    double salary = Double.parseDouble(scanner.nextLine());  
}  
System.out.println(salary);    // Error!
```



# Условни конструкции

## Решаване на задачи в клас (лаб)

- Напишете програма, която:
  - Прочита **вид** на **геометрична фигура** ("square", "rectangle", "circle" или "triangle")
  - Пресмята **лицето** спрямо вида на фигурата
- Примерен вход и изход:

square  
5



25

rectangle  
7  
2.5



17.5



```
String shape = scanner.nextLine();
double area = 0.0;
if(shape.equals("square")) {
    double side = Double.parseDouble(scanner.nextLine());
    area = side * side;
} else if(shape.equals("rectangle")) {
    double sideA = Double.parseDouble(scanner.nextLine());
    double sideB = Double.parseDouble(scanner.nextLine());
    area = sideA * sideB;
} //TODO: add more conditions
System.out.println(area);
```



# Дебъгване

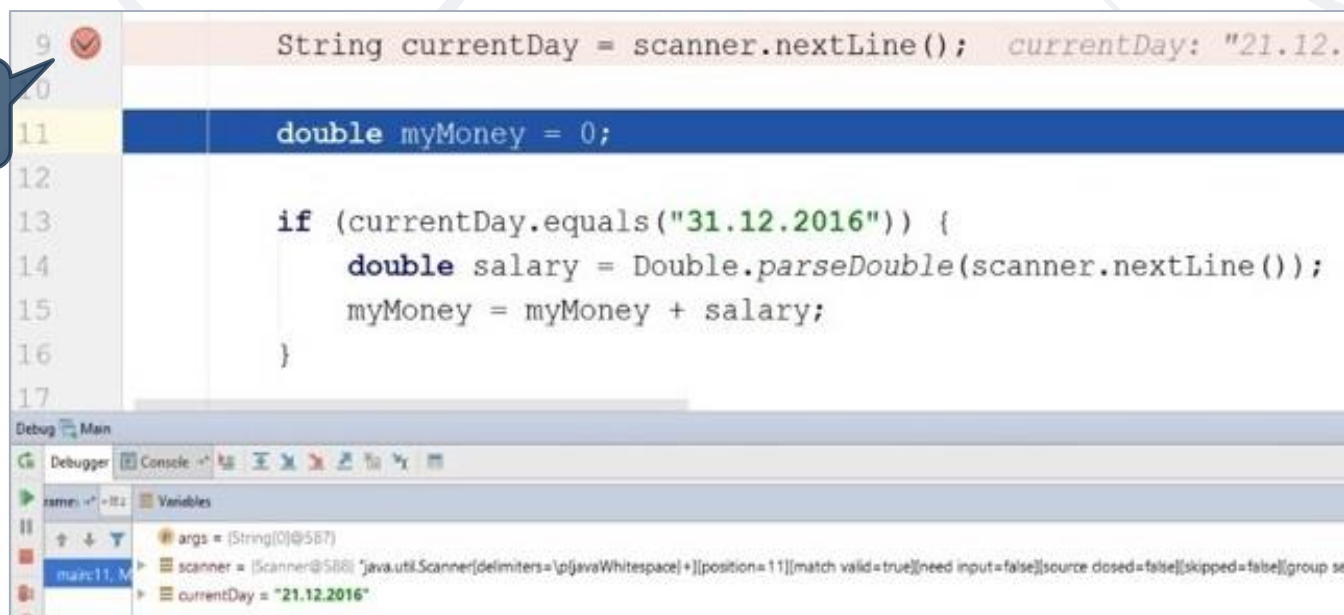
Прости операции с дебъгер

# Дебъгване

- Процес на проследяване на изпълнението на програмата
  - Това ни позволява да откриваме грешки (бъгове)



Breakpoint



```
9 String currentDay = scanner.nextLine(); currentDay: "21.12.  
10  
11 double myMoney = 0;  
12  
13 if (currentDay.equals("31.12.2016")) {  
14     double salary = Double.parseDouble(scanner.nextLine());  
15     myMoney = myMoney + salary;  
16 }  
17
```

Debug | Main

Debugger | Console | Variables

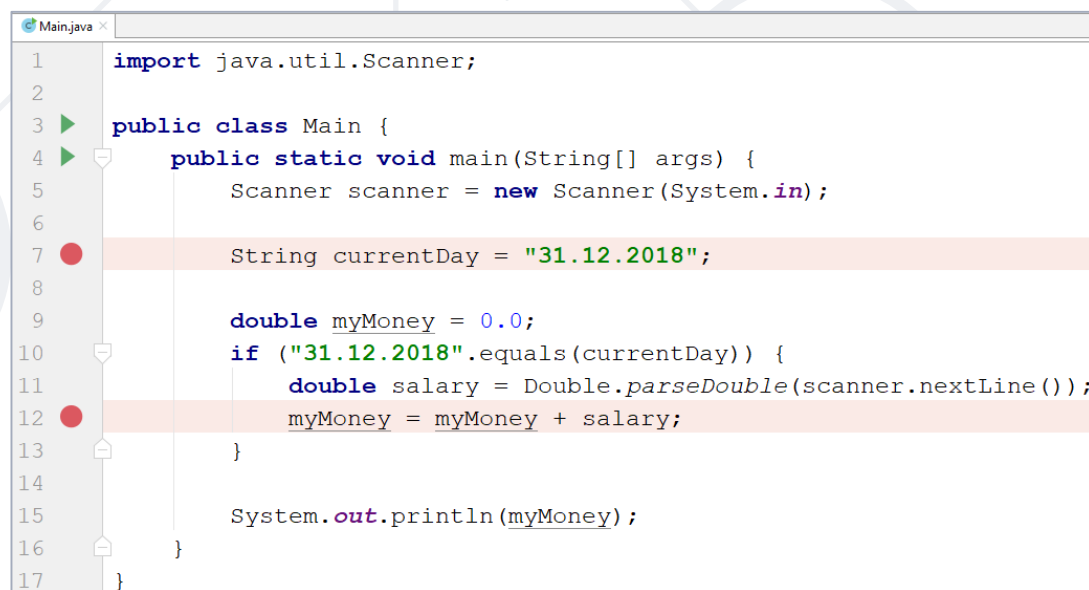

name: "args" | value: (String[0]@567)

name: "scanner" | value: (Scanner@568) java.util.Scanner[delimiters="/p/javaWhitespace"] [position=11][match valid=true][need input=false][source closed=false][skipped=false][group se

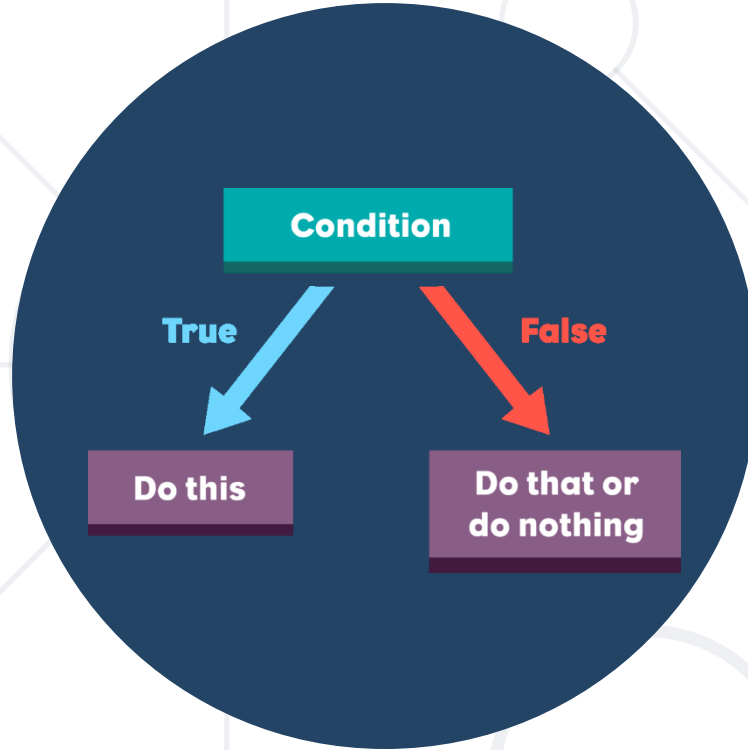
name: "currentDay" | value: "21.12.2016"

# Дебъгване във IntelliJ IDEA

- Натискане на **[Shift + F9]** ще стартира програмата в debug режим
- Можем да преминем към следващата стъпка с **[F8]**
- Можем да създаваме **[Ctrl + F8]** стопери – breakpoints
  - До тях можем директно да стигнем използвайки **[F9]**



```
1  import java.util.Scanner;
2
3  public class Main {
4      public static void main(String[] args) {
5          Scanner scanner = new Scanner(System.in);
6
7          String currentDay = "31.12.2018";
8
9          double myMoney = 0.0;
10         if ("31.12.2018".equals(currentDay)) {
11             double salary = Double.parseDouble(scanner.nextLine());
12             myMoney = myMoney + salary;
13         }
14
15         System.out.println(myMoney);
16     }
17 }
```



# Условни конструкции

## Решаване на задачи в клас (лаб)

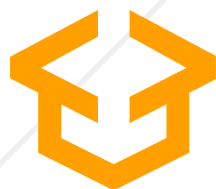
- Конструкции за проверка на условие – **if** и **if-else**
- Живот на променливата
- Дебъгване



# Въпроси?



**SoftUni**



**Software  
University**



**SoftUni  
Svetlina**



**SoftUni  
Creative**



**SoftUni  
Digital**



**SoftUni  
Foundation**



**SoftUni  
Kids**

# SoftUni Diamond Partners



**XS**software



**SBTech**  
*we know sports*



telenor



**SoftwareGroup**  
*doing it right*

**NETPEAK**



**SmartIT**



**Postbank**

*Решения за твоето утре*



**INDEAVR**

*Serving the high achievers*



**INFRAGISTICS®**



**STEMO®**  
*Computer Systems & Software*

**SUPERHOSTING.BG**



# SoftUni Organizational Partners

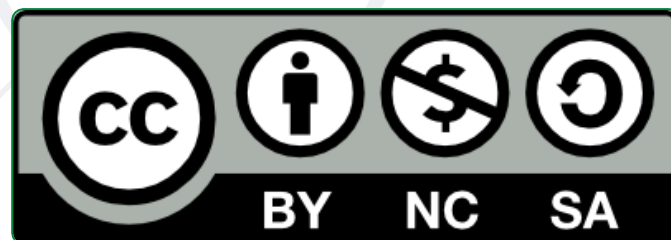


OneBit  
SOFTWARE



WORLD  
OF  
MYTHS

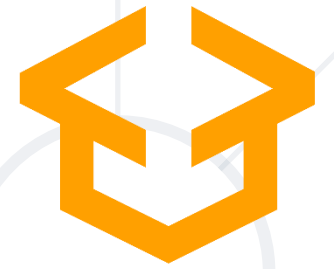
- Настоящият курс (слайдове, примери, видео, задачи и др.) се разпространяват под свободен лиценз "Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International"



- Благодарности: настоящият материал може да съдържа части от следните източници
  - Книга "Основи на програмирането с Java" от Светлин Наков и колектив с лиценз CC-BY-SA

# Обучения в СофтУни

- Software University – High-Quality Education, Profession and Job for Software Developers
  - [softuni.bg](http://softuni.bg)
- Software University Foundation
  - <http://softuni.foundation/>
- Software University @ Facebook
  - [facebook.com/SoftwareUniversity](https://facebook.com/SoftwareUniversity)
- Software University Forums
  - [forum.softuni.bg](http://forum.softuni.bg)



**Software  
University**

