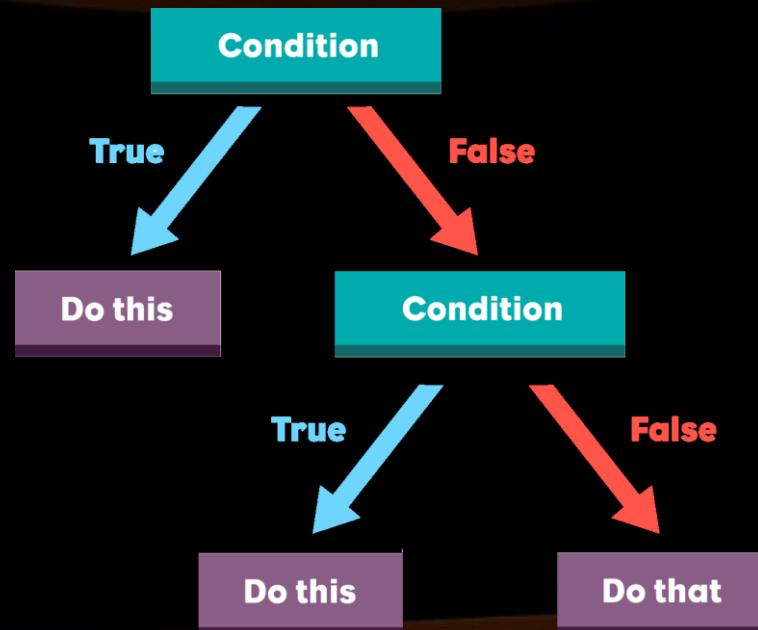


# По-сложни проверки

Вложени if конструкции и  
по-сложни логически условия



Софтуни  
трейнърски екип  
Софтуерен университет  
<http://softuni.bg>



Have a Question?



sli.do

#TODO

# Съдържание

1. Вложени проверки
2. Сложни проверки
  - Логическо "и", "или", отрицание и приоритет на условия
3. Switch конструкция



```
if (condition)
```

## Вложени проверки

If-конструкции, вложени една в друга

# Вложени проверки

- Конструкциите **if-else** могат да се влагат една в друга:

```
if (condition1) {  
    console.log("condition1 valid");  
    if (condition2)  
        console.log("condition2 valid");  
    else  
        console.log("condition2 not valid");  
}
```

Вложена if  
конструкция

- Само при **изпълнение** на първото условие се преминава към вложената проверка

# Обръщение според възраст и пол – условие

- Напишете програма, която:
  - Чете от потребителя възраст и пол:
  - Принтира обръщение според въведеното, както е показано на схемата (в следващия слайд)
- Примерен вход и изход:

Вход:      Изход:

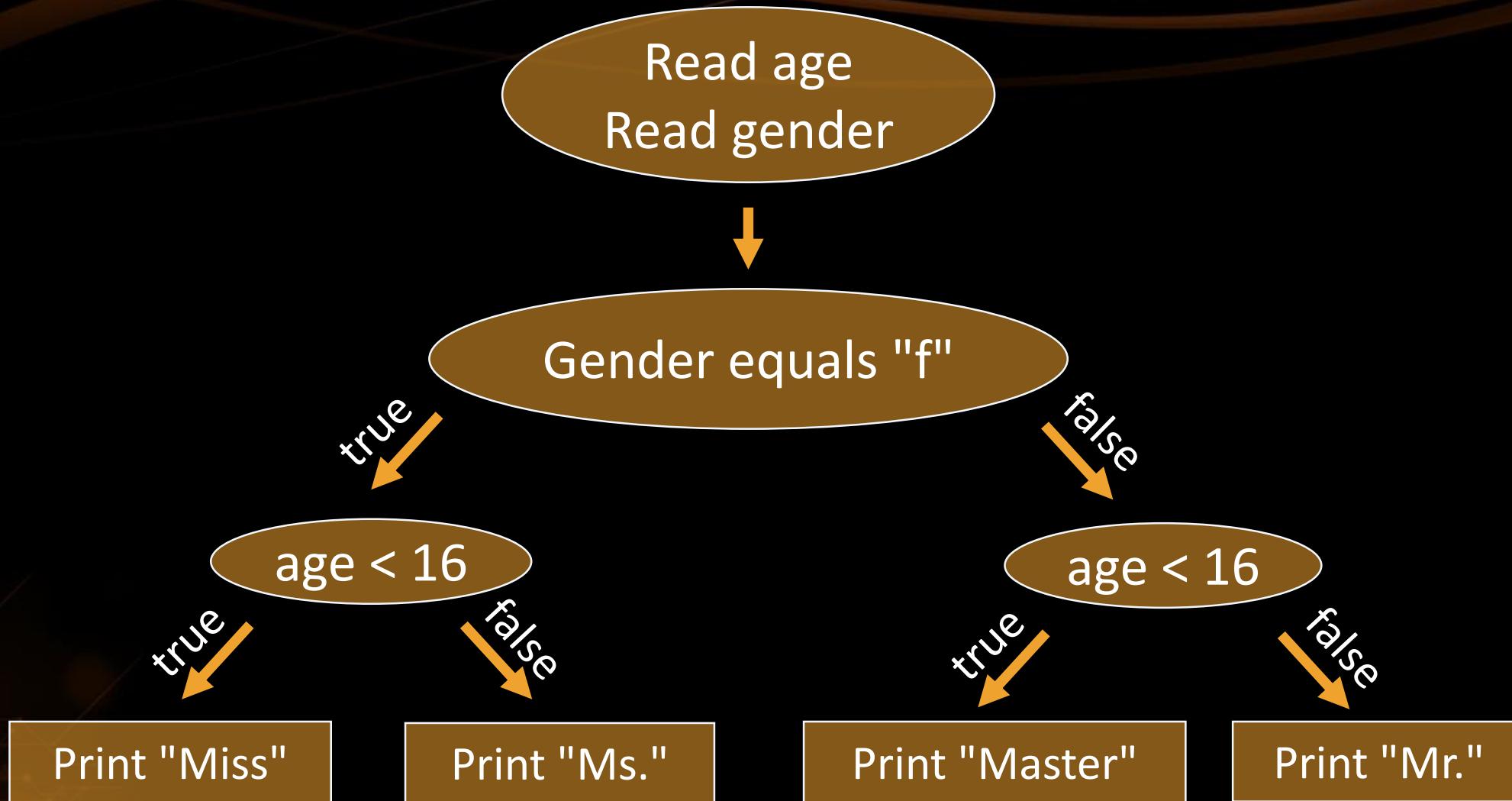
|    |      |
|----|------|
| 12 | Miss |
| f  |      |

Вход:      Изход:

|    |     |
|----|-----|
| 16 | Mr. |
| m  |     |



# Обръщение според възраст и пол – условие (2)



# Решение: Обръщение според възраст и пол



```
function personalTitles([arg1, arg2]) {  
    let age = Number(arg1);  
    let gender = arg2;  
    if (gender == "f") {  
        if (age < 16) console.log("Miss");  
        else console.log("Ms.");  
    } else if (gender == "m") {  
        if (age < 16) console.log("Master");  
        else console.log("Mr.");  
    }  
}
```

Тестване на решението: <https://judge.softuni.bg/Contests/Practice/Index/153#0>

# Квартално магазинче – условие

- Напишете програма, която:
  - Чете от потребителя **име на продукт, град, количество**:
  - Пресмята **цената на продукта** спрямо таблицата:

| град / продукт | coffee | water | beer | sweets | peanuts |
|----------------|--------|-------|------|--------|---------|
| Sofia          | 0.50   | 0.80  | 1.20 | 1.45   | 1.60    |
| Plovdiv        | 0.40   | 0.70  | 1.15 | 1.30   | 1.50    |
| Varna          | 0.45   | 0.70  | 1.10 | 1.35   | 1.55    |

# Квартално магазинче – условие (2)

- Примерен вход и изход:

peanuts  
Plovdiv  
1

1.5

coffee  
Varna  
2

0.9

beer  
Sofia  
6

7.2

| град / продукт | coffee | water | beer | sweets | peanuts |
|----------------|--------|-------|------|--------|---------|
| Sofia          | 0.50   | 0.80  | 1.20 | 1.45   | 1.60    |
| Plovdiv        | 0.40   | 0.70  | 1.15 | 1.30   | 1.50    |
| Varna          | 0.45   | 0.70  | 1.10 | 1.35   | 1.55    |

# Решение: Квартално магазинче

```
function smallShop([arg1, arg2, arg3]) {  
    let product = arg1.toLowerCase();  
    let town = arg2.toLowerCase();  
    let quantity = Number(arg3);  
    if (town == "sofia") {  
        if (product == "coffee")  
            console.log(0.50 * quantity);  
        // TODO: Check the other cases ...  
    }  
    if (town == "varna") // TODO: finish this ...  
    if (town == "plovdiv") // TODO: finish this ...  
}
```

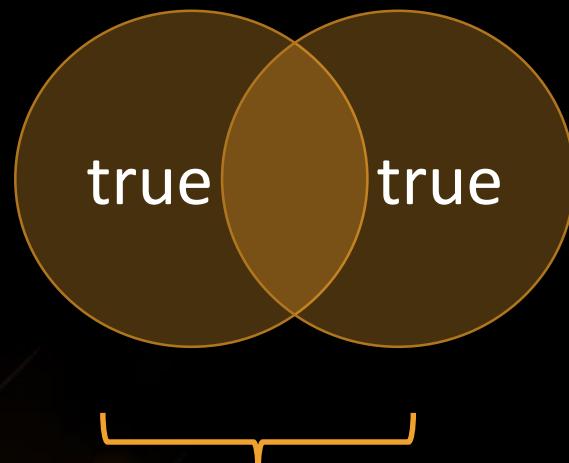
Тестване на решението: <https://judge.softuni.bg/Contests/Practice/Index/153#1>

```
if (a || b)  
else if (!c && d)
```

**По-сложни проверки**  
**Булеви оператори**

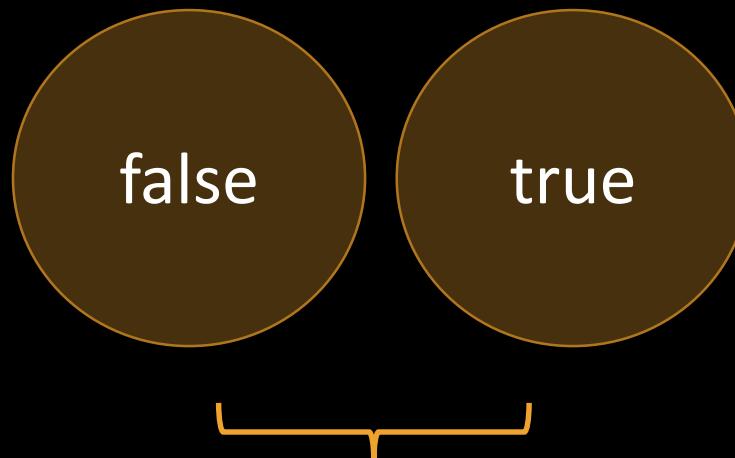
- Оператори, които комбинират или изключват условия
- Връщат булев резултат (true или false)

"`&&`" - И



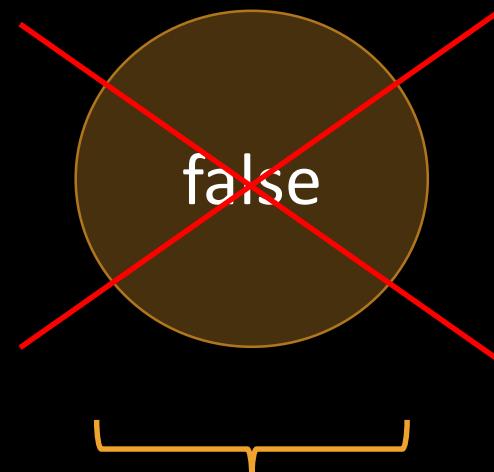
Вярност на двете условия

"`||`" - ИЛИ



Вярност на  
едното или другото  
условие

"`!`" - ОТРИЦАНИЕ



Отрицание на условие

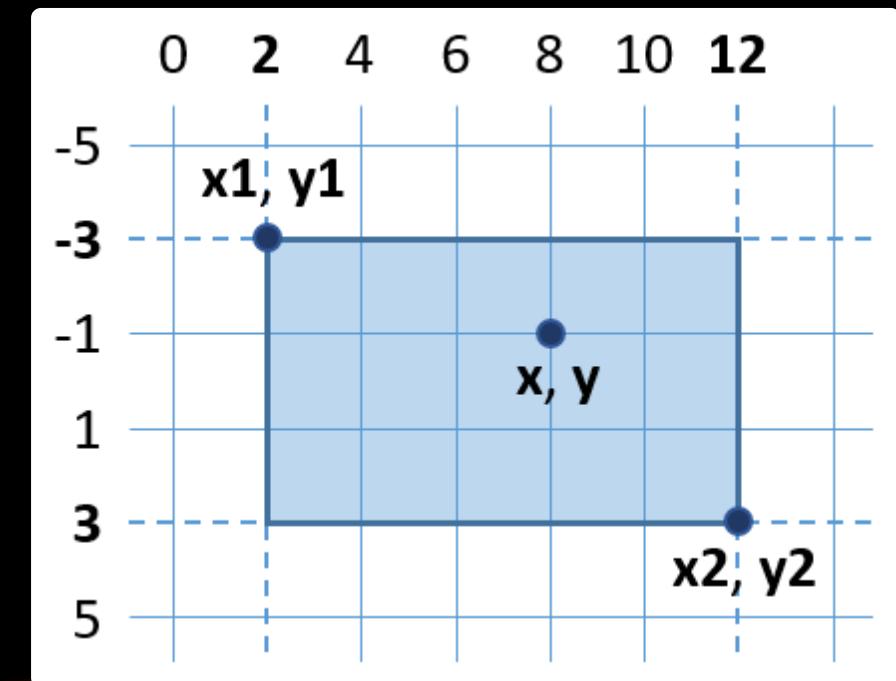
- Проверява изпълнението на няколко условия едновременно
- Пример: проверка дали число е едновременно:
  - по-голямо от 5 и по-малко от 10
  - четно

```
let a = parseInt(arg1);
if (a > 5 && x < 10 && a % 2 == 0) ...
```

&&

# Точка в правоъгълник - условие

- Напишете програма, която:
  - Проверява дали точка е вътрешна за даден правоъгълник
- Точка е вътрешна, ако е едновременно:
  - Надясно от лявата му страна
  - Наляво от дясната му страна
  - Надолу от горната му страна
  - Нагоре от долната му страна



# Точка в правоъгълник - решение

```
function pointInRectangle([arg1, arg2, arg3, arg4, arg5, arg6]) {  
    // TODO: Read the coordinates of the points ...  
    let x = Number(arg5), y = Number(arg6);  
    if (x >= x1 && x <= x2 && y >= y1 && y <= y2) {  
        console.log("Inside");  
    }  
    else {  
        console.log("Outside");  
    }  
}  
pointInRectangle([2, -3, 12, 3, 8, -1]); ➔
```

DEBUG CONSOLE  
Debugging with inspector protocol  
node complexConditions.js  
Inside

Тестване на решението: <https://judge.softuni.bg/Contests/Practice/Index/153#2>

- Проверява дали е изпълнено поне едно между няколко условия
- Пример: проверка дали въведената дума е:
  - "Example" или "Demo"

```
let word = arg1;  
if (word == "Example" || word == "Demo") ...
```

# Плод или зеленчук - условие

- Напишете програма, която:
  - Проверява дали въведеният вход от потребителя е плод или зеленчук между изброените:
    - Плодове: banana, apple, kiwi, cherry, lemon, grapes
    - Зеленчуци: tomato, cucumber, pepper, carrot
    - Извежда "vegetable", "fruit" или "unknown"
  - Примерен вход и изход:

lemon



fruit

carrot



vegetable

java



unknown



# Плод или зеленчук - решение

```
function fruitOrVegetable([arg1]) {  
    let p = arg1;  
    if (p == "banana" || p == "apple" || p == "kiwi" ||  
        p == "cherry" || p == "lemon" || p == "grapes")  
        console.log("fruit");  
    else if (p == "tomato" || p == "cucumber" ||  
             p == "pepper" || p == "carrot")  
        console.log("vegetable");  
    else  
        console.log("unknown");  
}  
fruitOrVegetable(["banana"]); ➔
```

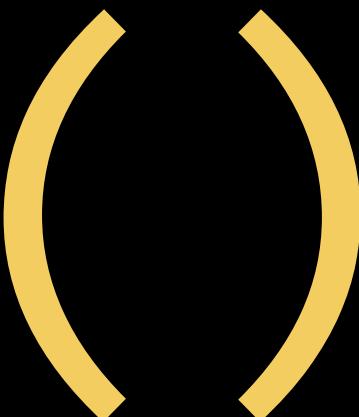
DEBUG CONSOLE

Debugging with inspector protocol  
node complexConditions.js  
fruit

Тестване на решението: <https://judge.softuni.bg/Contests/Practice/Index/153#3>

- Чрез скоби ( ) можем да приоритизираме условия
- Пример:
  - Проверка дали число е в диапазона [100... 200] или е равно на 0

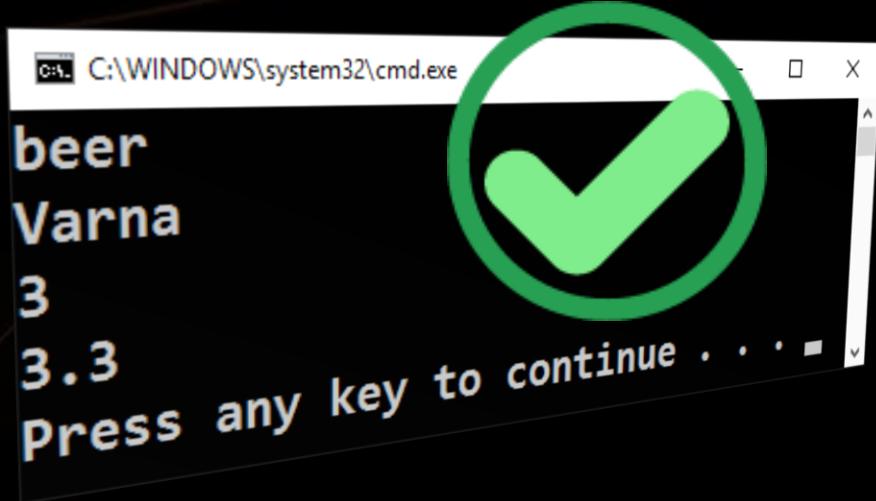
```
let a = 5;
if ((a >= 100 && a <= 200) || a == 0) {
    console.log("In range");
}
```



- Проверява дали **не е изпълнено** дадено условие
- Пример:
  - Проверка дали число е **по-голямо** от **10** и е **четно**

```
let isValid = (number > 10) && (number % 2 == 0);
if (!isValid) {
    console.log("Invalid");
}
```



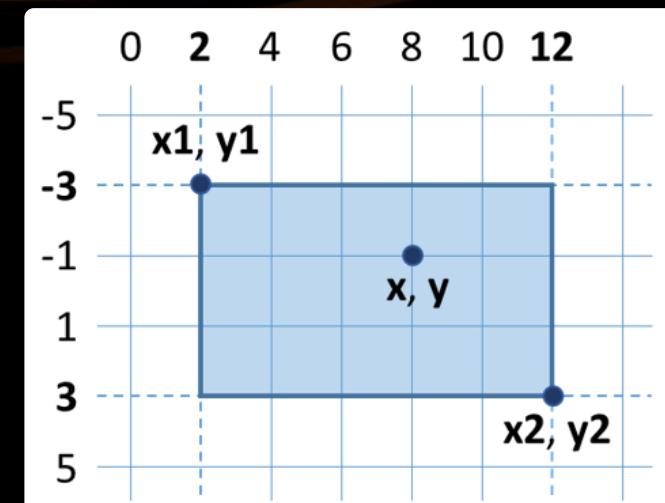


# По-сложни проверки

Работа на живо в клас (лаб)

# Точка върху страна на правоъгълник - условие

- Напишете програма, която:
  - Чете 6 десетични числа  $x_1, y_1, x_2, y_2, x$  и  $y$
  - Извежда дали точката е:
    - Върху страна от правоъгълника ("Border")
    - В или извън правоъгълника ("Inside/Outside")



■ Примерен  
вход и изход:

|    |
|----|
| 2  |
| -3 |
| 12 |
| 3  |
| 8  |
| -1 |



Inside /  
Outside

|    |
|----|
| 2  |
| -3 |
| 12 |
| 3  |
| 12 |
| -1 |

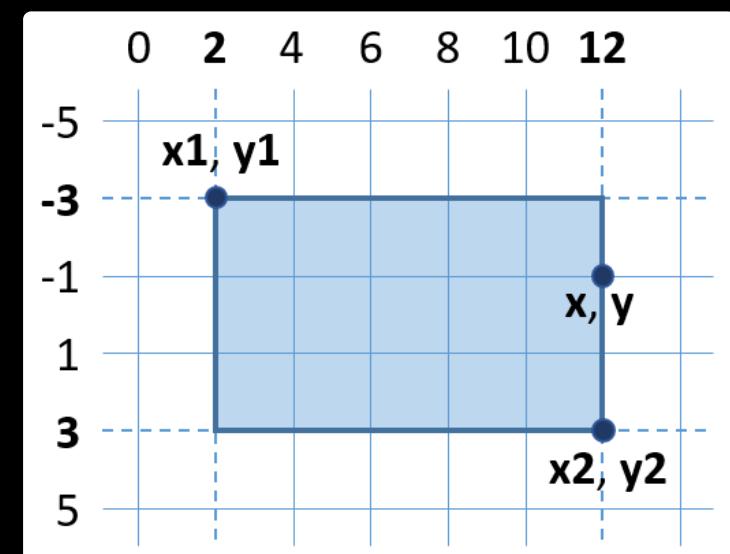


Border

# Точка върху страна на правоъгълник - решение

- Точка лежи върху някоя от страните на правоъгълник, ако:
  - $x$  съвпада с  $x_1$  или  $x_2$  и същевременно  $y$  е между  $y_1$  и  $y_2$  или
  - $y$  съвпада с  $y_1$  или  $y_2$  и същевременно  $x$  е между  $x_1$  и  $x_2$

```
if (((x == x1 || x == x2) &&
    (y >= y1) && (y <= y2)) ||
    ((y == y1 || y == y2) &&
    (x >= x1) && (x <= x2))) {
    console.log("Border");
}
```



# Опростяване на логически условия

- Предходното условие може да се опости ето така:

```
function isOnBorder([arg1, arg2, arg3, arg4, arg5, arg6]) {  
    // TODO: Read the input points  
    let onLeftSide = (x == x1) && (y >= y1) && (y <= y2);  
    let onRightSide = (x == x2) && (y >= y1) && (y <= y2);  
    let onUpSide = (y == y1) && (x >= x1) && (x <= x2);  
    let onDownSide = (y == y2) && (x >= x1) && (x <= x2);  
  
    if (onLeftSide || onRightSide || onUpSide || onDownSide)  
        console.log("Border");  
    else  
        console.log("Inside / Outside");  
}
```

Тестване на решението: <https://judge.softuni.bg/Contests/Practice/Index/153#5>

# Магазин за плодове – условие

- Напишете програма, която:
  - Чете потребителски вход: продукт, ден, количество
  - Извежда сумата, която трябва да се заплати според дня и продукта



# Магазин за плодове – условие (2)

- В работни дни продава на следните цени:

| плод | banana | apple | orange | grapefruit | kiwi | pineapple | grapes |
|------|--------|-------|--------|------------|------|-----------|--------|
| цена | 2.50   | 1.20  | 0.85   | 1.45       | 2.70 | 5.50      | 3.85   |

- В почивни дни цените са по-високи:

| плод | banana | apple | orange | grapefruit | kiwi | pineapple | grapes |
|------|--------|-------|--------|------------|------|-----------|--------|
| цена | 2.70   | 1.25  | 0.90   | 1.60       | 3.00 | 5.60      | 4.20   |

- Примерен вход и изход:



# Магазин за плодове - решение

```
function fruitShop([arg1, arg2, arg3]) {  
    let fruit = arg1;  
    let day = arg2;  
    let quantity = Number(arg3);  
    if (day == "saturday" || day == "sunday") {  
        if (fruit == "banana") price = 2.70;  
        else if (fruit == "apple") price = 1.25;  
        // TODO: more fruits come here ...  
    }  
    else if (day == "monday" || day == "tuesday" || day ==  
            "wednesday" || day == "thursday" || day == "friday") {  
        if (fruit == "banana") price = 2.50;  
        // TODO: more fruits come here ...  
    }  
}
```

Тестване на решението: <https://judge.softuni.bg/Contests/Practice/Index/153#6>

# Търговски комисионни - условие

- Напишете програма, която:
  - Чете от потребителя: град, обем на продажби (десетично число)
  - Изчислява комисионната, която дадена фирма дава на търговците според града и обема на продажбите
  - Извежда стойността на комисионната, закръглена до 2 цифри след десетичната запетая



# Търговски комисионни – условие (2)

| Град / цена | $0 \leq s \leq 500$ | $500 < s \leq 1\ 000$ | $1\ 000 < s \leq 10\ 000$ | $s > 10\ 000$ |
|-------------|---------------------|-----------------------|---------------------------|---------------|
| Sofia       | 5%                  | 7%                    | 8%                        | 12%           |
| Varna       | 4.5%                | 7.5%                  | 10%                       | 13%           |
| Plovdiv     | 5.5%                | 8%                    | 12%                       | 14.5%         |

- Примерен вход и изход:

Plovdiv  
499.99



27.50

Sofia  
12 000



1 440

# Търговски комисионни - решение

```
function tradeComissions([arg1, arg2]) {  
    let town = arg1.toLowerCase();  
    let sales = Number(arg2);  
    let comission = -1.0;  
    if (town == "sofia") {  
        if (0 <= sales && sales <= 500) comission = 0.05;  
        else if (500 < sales && sales <= 1000) comission = 0.07;  
        // TODO: check the other price ranges ...  
    }  
    else if (town == "varna") // TODO: check the price ranges ...  
    else if (town == "plovdiv") // TODO: check the price ranges ...  
  
    if (comission >= 0)  
        console.log((sales * comission).toFixed(2));  
    else console.log("error");  
}
```

Тестване на решението: <https://judge.softuni.bg/Contests/Practice/Index/153#7>

```
switch (value)
```

```
case a: ...
```

```
case b: ...
```

# Условна конструкция Switch

По-доброто If-Else-If-Else?

# Условна конструкция switch

- Работи като поредица if/else-if/else-if...

Условието в  
switch-case е  
стойност

default  
работи като  
else

```
switch (...) {  
    case ...:  
        // code  
        break;  
    case ...:  
        // code  
        break;  
    default:  
        // code  
        break;  
}
```

```
let value = ...  
if (value == ...)  
    // code  
else  
    // code
```

# Ден от седмицата - условие

- Напишете програма, която:
  - Чете цяло число, въведено от потребителя
  - Извежда **дения от седмицата** с текст (на английски), според въведеното число (1...7)
  - Извежда "Error!", ако числото **не е в диапазона**
- Примерен вход и изход:



# Ден от седмицата - решение

```
function dayOfWeek([arg1]) {  
    let day = parseInt(arg1);  
    switch (day) {  
        case 1: console.log("Monday"); break;  
        case 2: console.log("Tuesday"); break;  
        //TODO: more cases come here  
        case 7: console.log("Sunday"); break;  
        default: console.log("Error!"); break;  
    }  
}  
  
dayOfWeek([8]);
```



DEBUG CONSOLE

Debugging with inspector protocol  
node complexConditions.js  
Error!

Тестване на решението : <https://judge.softuni.bg/Contests/Practice/Index/153#8>

# Множество случаи в switch

- Чрез **switch**, можем да изпълняваме един и същ код за множество условия

Кодът ще се  
изпълни за някое  
от трите условия в  
серията

```
switch (...) {  
    case ... :  
    case ... :  
    case ... :  
        // code  
        break;  
  
    default:  
        // code  
        break;  
}
```

```
let value = ...  
if (value == ... ||  
    value == ... ||  
    value == ... ||)  
    // code  
else  
    // code
```

# Вид животно - условие

- Напишете програма, която:
  - Чете от потребителя дума (животно)
  - Извежда вида на животно според името му
    - Бозайник – "mammal"
    - Влечуго – "reptile"
    - Други – "unknown"
- Примерен вход и изход:

dog → mammal

car → unknown



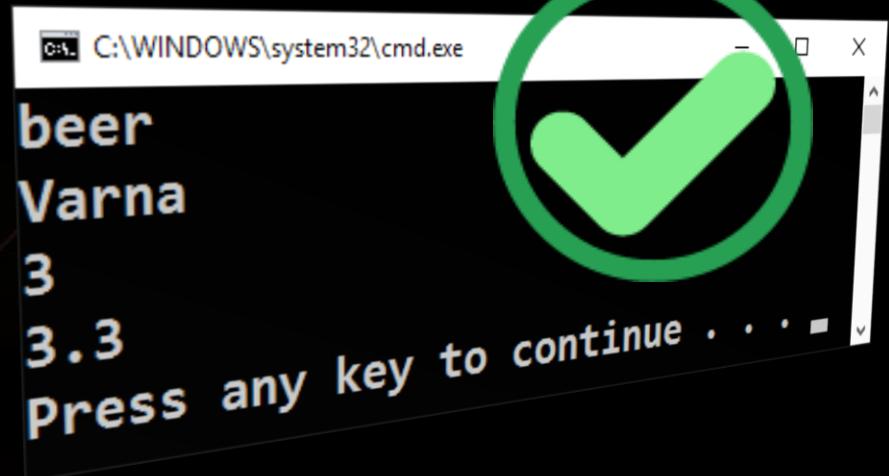
# Вид животно - решение

```
function animalType([arg1]) {  
    let animal = arg1;  
    switch (animal) {  
        case "dog": console.log("mammal"); break;  
        case "crocodile":  
        case "tortoise":  
        case "snake": console.log("reptile"); break;  
        default: console.log("unknown"); break;  
    }  
}  
  
animalType(["snake"]);
```



## DEBUG CONSOLE

Debugging with inspector protocol  
node complexConditions.js  
reptile



# По-сложни проверки

Работа на живо в клас (лаб)

# Какво научихме днес?

- По-сложни проверки с **&&**, **||**, **!** и **()**

```
if (condition && condition)
    // code
else if (condition || condition)
    // code
else if (!condition)
    // code
else
    // code
```



# Какво научихме днес? (2)

- Switch-case конструкция и вложени проверки

```
switch (...) {  
    case ...:  
        //code  
        break;  
    case ...:  
        //code  
        break;  
    default:  
        //code  
        break;  
}
```

```
if (condition) {  
    if (condition)  
        // code  
    else if (...)  
        // code  
    else  
        // code  
}
```



# По-сложни проверки



## Въпроси?

# SUPERHOSTING.BG



- 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 със C#" от Светлин Наков и колектив с лиценз CC-BY-SA

# Бесплатни обучения в СофтУни



- Фондация "Софтуерен университет" – [softuni.org](http://softuni.org)
- Софтуерен университет – качествено образование, професия и работа за софтуерни инженери
  - [softuni.bg](http://softuni.bg)
- СофтУни @ Facebook
  - [facebook.com/SoftwareUniversity](https://facebook.com/SoftwareUniversity)
- СофтУни форуми – [forum.softuni.bg](http://forum.softuni.bg)

