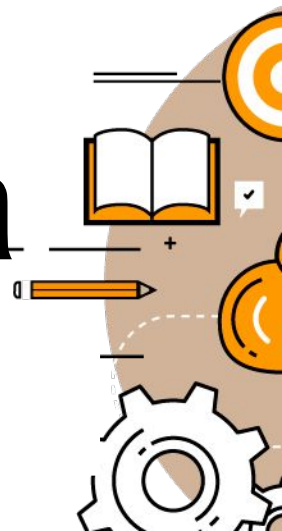
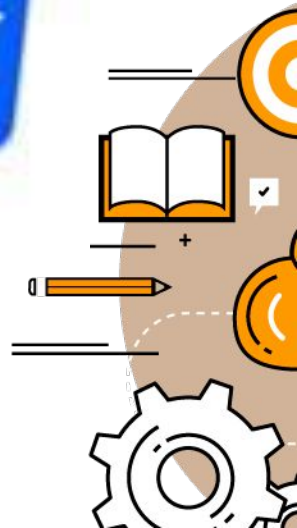




Основные конструкции языка Java



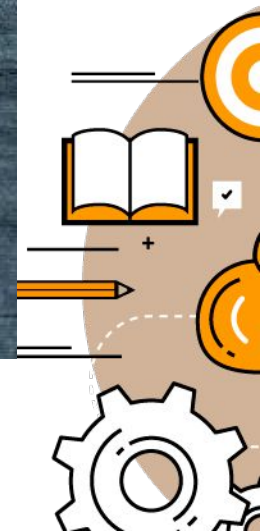


Существует три типа комментариев:

- **Если ваш комментарий укладывается в одну строку, начните его с двух косых черт:**
`// Этот метод вычисляет расстояние`
- **Более длинные многострочные комментарии должны быть окружены символами `/*` и `*/`, например:**
`/* следующие 3 строки кода
нужны для сохранения позиции рыбы.
*/`
- **Вместе с Java поставляется утилита `javadoc`, которая позволяет извлечь комментарии из вашей программы в отдельный файл справки. Этот файл может быть использован в качестве технической документации для ваших программ. Такие комментарии должны находиться между символами `/**` и `*/`. Только самые важные комментарии, такие как описание класса или метода, следует помещать между этими символами.**
`/** Этот метод вычисляет размер скидки в зависимости от цены. Если цена больше $100, скидка = 20%, в противном случае только 10%.
*/`

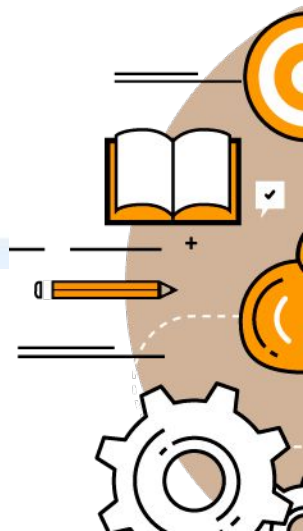


Выбор



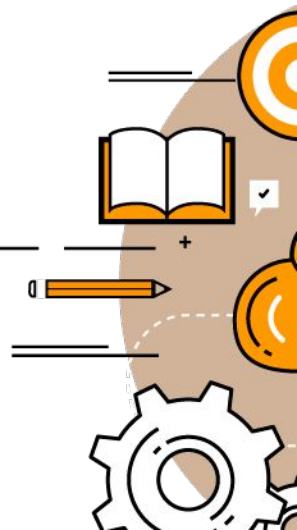
Выбираем какую скидку нам давать покупателю

```
2
3 public class NewApp {
4     public static void main(String[] args) {
5         double price = 80;
6         // Более дорогие товары продаются со скидкой 20%
7         if (price > 100)
8         {
9             price=price*0.8;
10            System.out.println("Ваша скидка 20%");
11        }
12        else
13        {
14            price=price*0.9;
15            System.out.println("Ваша скидка 10%");
16        }
17    }
18 }
19
```



Усовершенствуем наш класс Fish, добавим туда условие

```
2
3 public class Fish extends Pet{
4     int currentDepth=0;
5     public int dive(int howDeep)
6     {
7         currentDepth=currentDepth + howDeep;
8         if (currentDepth > 100){
9             System.out.println("Я маленькая рыбка "+
10                " и не могу плавать глубже 100 метров");
11             currentDepth=currentDepth - howDeep;
12         } else {
13             System.out.println("Погружаюсь ещё на " + howDeep +
14                " метров");
15             System.out.println("Я на глубине " + currentDepth +
16                " метров");
17         }
18         return currentDepth;
19     }
20     public String say(String something)
21     {
22         return "Ты чё не знаешь, что рыбы не разговаривают?";
23     }
24 }
25
```



```
2
3 public class NewApp {
4     public static void main(String[] args) {
5         Fish myFish = new Fish();
6         // Попробуем заставить рыбу погрузиться ниже 100 метров
7         myFish.dive(2);
8         myFish.dive(97);
9         myFish.dive(3);
10    }
11 }
12
```

Problems @ Javadoc Declaration Console

<terminated> NewApp [Java Application] C:\Program Files\Java\jre1.8.0_151\bin\javaw.exe (2 ф

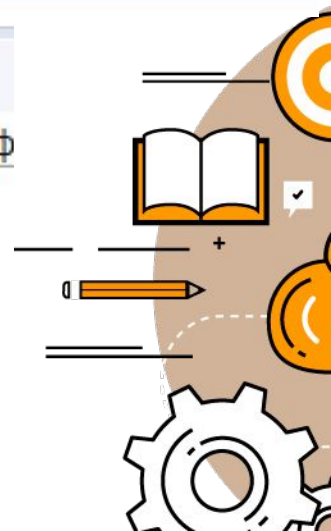
Погружаюсь ещё на 2 метров

Я на глубине 2 метров

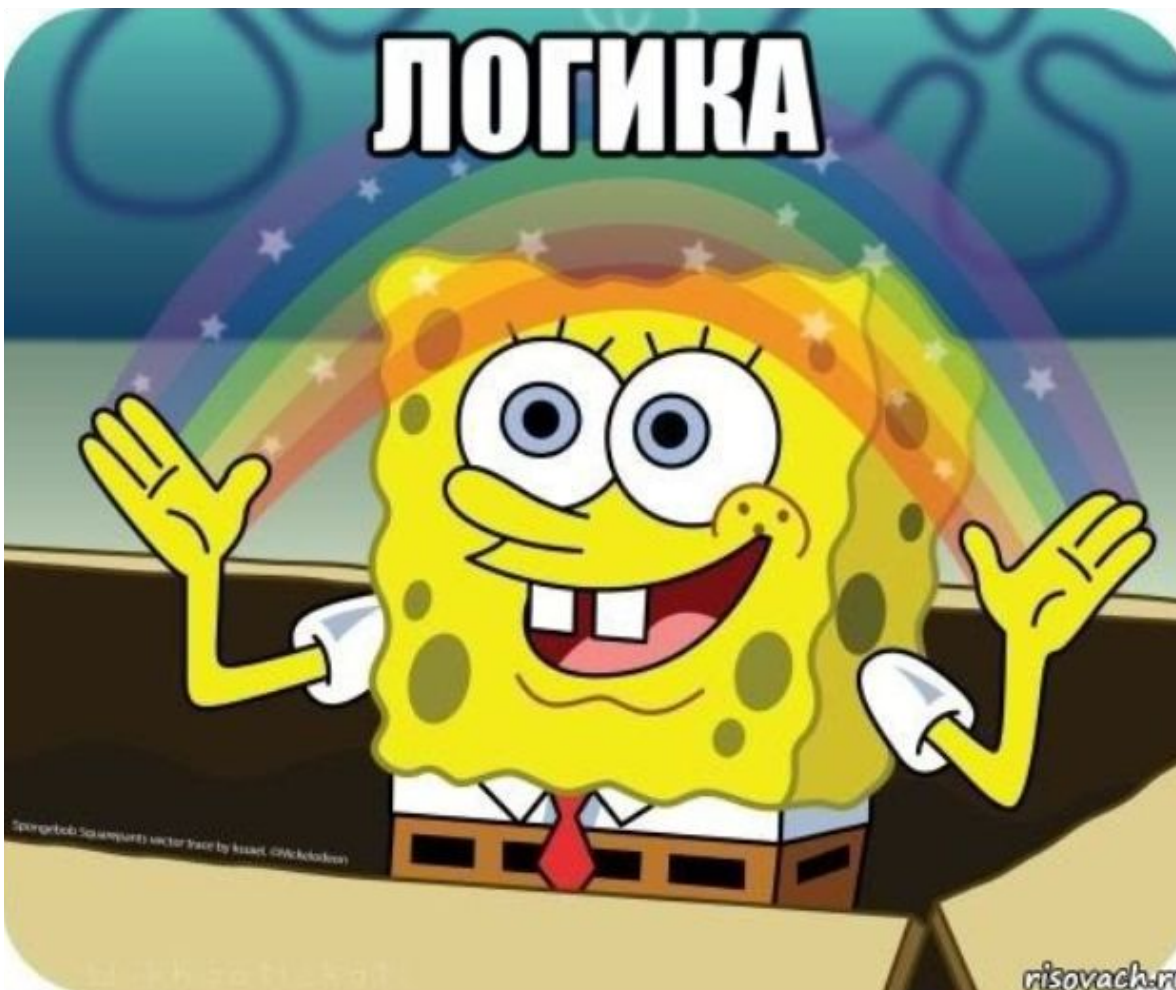
Погружаюсь ещё на 97 метров

Я на глубине 99 метров

Я маленькая рыбка и не могу плавать глубже 100 метров



ЛОГИКА



SpongeBob SquarePants vector trace by Asael. ©Nickelodeon

risovach.ru

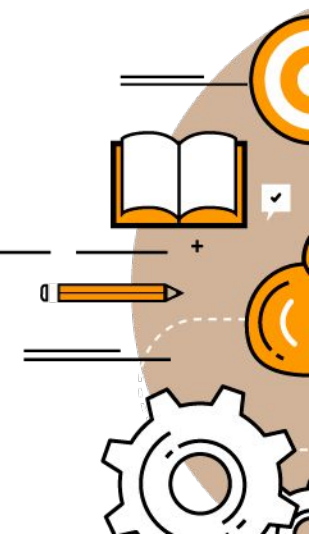
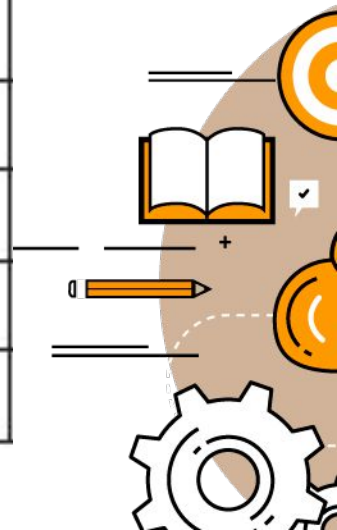


Таблица истинности

Логические переменные		отрицание	конъюнкция	дизъюнкция
A	B	$\neg A$	$A \& B$	$A \vee B$
0	0	1	0	0
0	1	1	0	1
1	0	0	0	1
1	1	0	1	1



Использование логического «ИЛИ» и «И»

```
if (state.equals("Texas") | state.equals("California"))
```

Можно также использовать две вертикальные линии в операторе if:

```
if (state.equals("Texas") || state.equals("California"))
```

```
if (state.equals("New York") && price >110)
```

ИЛИ

```
if (state.equals("New York") & price >110)
```



Логическое «не»

```
if (!state.equals("New York"))
```

Другой пример – следующие выражения абсолютно идентичны:

```
if (price < 50)
```

```
if (!(price >=50))
```



Домашняя работа

- Создать класс прямоугольник(поля ширина и высота)
- Написать для него конструктор
- Написать функцию нахождения периметра окружности.
- Написать функцию которая, если ширина больше чем длина выводит сумму длины и ширины, если ширина больше чем 10 но меньше чем 30, то вывести ширину, в противном случае вывести значение длины прямоугольника.

