

ЛЕКЦИЯ I © 2020 Нет Ит

JAVA OOP:

Абстрактность

Теодор Костадинов



SOFTWARE
ACADEMY



Съдържание

1. Интерфейси
2. Абстрактни класове
3. Абстрактност



Интерфейс



Какво е интерфейс?



- Дефинират списък от операции, методи, без да дефинират самите тях.
- Нещо като обещание/договор, че един клас ще има дадени методи.
- Интерфейса, като концепция представлява описание на видимата (публичната) част от нашата програма (тази която искаме да направим достъпна) за други части от програмата, други класове или други модули.
- Служи за координиране между различни софтуерни екипи:
 - Единият екип дефинира от какви методи има нужда
 - Другият екип имплементира методите
 - Първият екип ги използва, без да е необходимо да знае какво точно правят под повърхността



Как работи?

- Интерфейсите се имплементират от класовете, по същият начин както един клас може да наследи друг клас
- Клас, който имплементира даден интерфейс е задължен да имплементира методите, които ги има в интерфейса

```
public interface OperateCar {  
  
    int turn (Direction direction,  
             double radius,  
             double startSpeed,  
             double endSpeed);  
    int changeLanes (Direction direction,  
                    double startSpeed,  
                    double endSpeed);  
    int signalTurn (Direction direction,  
                   boolean signalOn);  
}
```

```
public class OperateBMW760i implements OperateCar {  
  
    @Override  
    int signalTurn(Direction direction, boolean signalOn) {  
        // code to turn BMW's LEFT turn indicator  
        lights on and off  
    }  
  
    ....  
}
```


Особености

- Един клас може да наследява множество интерфейси.
- Също така един клас може и да наследява друг клас и едновременно да имплементира интерфейс.
- Един интерфейс също така може да наследява друг интерфейс.
- Един интерфейс може да наследява неограничен брой други интерфейси.
- Ако клас имплементира интерфейс, той е длъжен да овъоррайдне всичките му методи, освен ако класа не е абстрактен.



Абстрактни класове



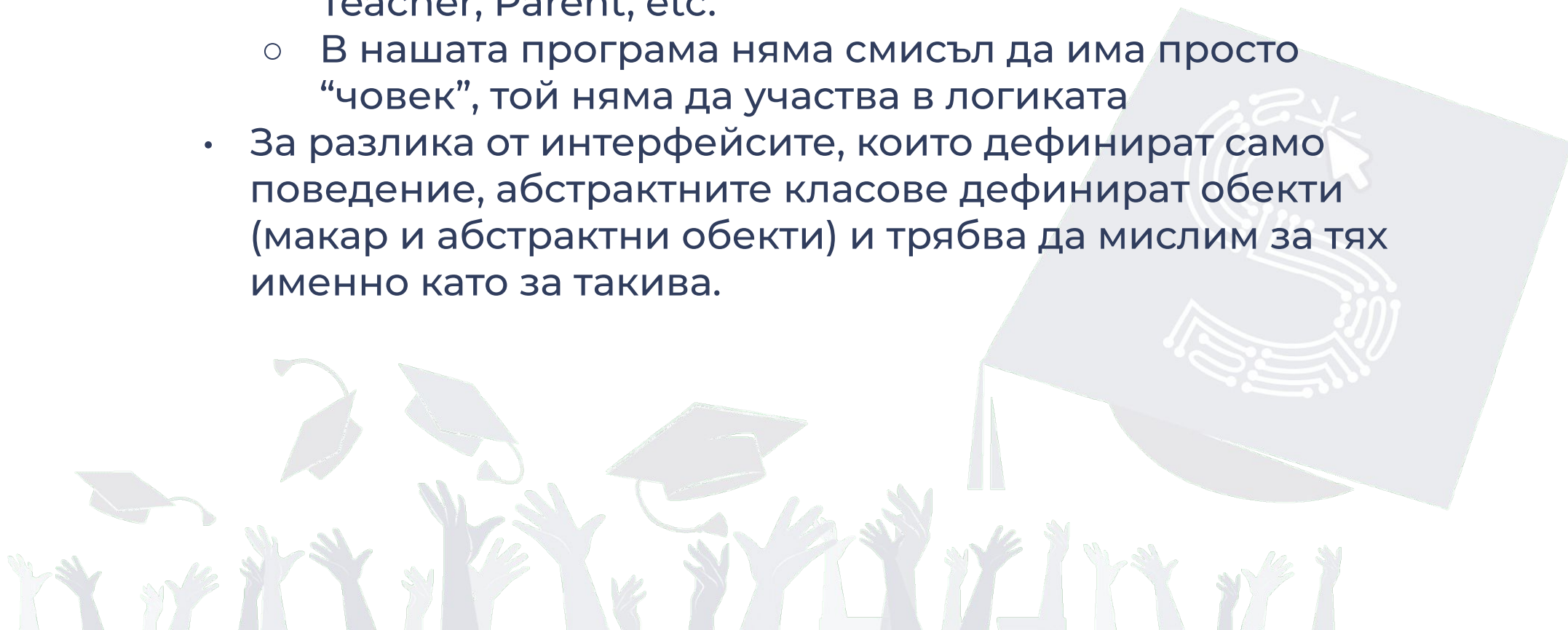
Особености

- Абстрактните класове са класове, които съдържат абстрактни методи.
- Абстрактните методи са методи без имплементация - само сигнатурата им е зададена. (подобно на методите в интерфейс)
- Това е възможно с добавянето на ключова дума **abstract** пред сигнатурата на метода и пред името на класа
- Не може да се създават обекти от абстрактни класове.
- Те могат само да бъдат наследявани, а абстрактните им методи - имплементирани, от класа наследник.



Смисъл

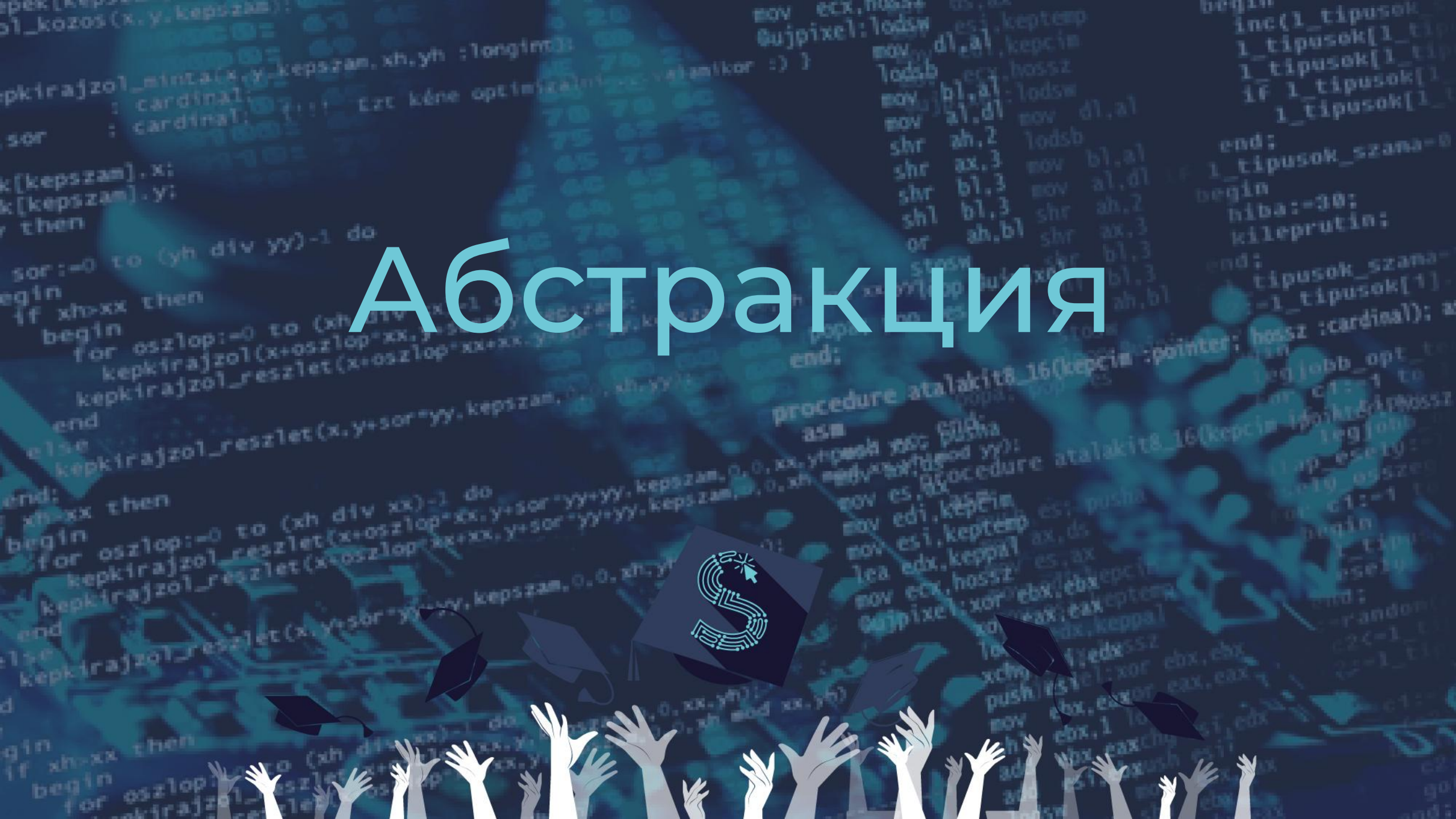
- Един клас има смисъл да е абстрактен, когато не искаме (няма причина) от него да се създават обекти. Например:
 - `class Human`, който е наследяван от клас `Student`, `Teacher`, `Parent`, etc.
 - В нашата програма няма смисъл да има просто “човек”, той няма да участва в логиката
- За разлика от интерфейсите, които дефинират само поведение, абстрактните класове дефинират обекти (макар и абстрактни обекти) и трябва да мислим за тях именно като за такива.



Абстрактни класове и Интерфейси

- Всеки метод в един интерфейс е `abstract public`, докато при абстрактните класове, методите могат да са `package private`, `protected` или `public`
- Всички методи, които са дефинирани в един интерфейс ще бъдат имплементирани, като публични, докато при абстрактният клас, наследникът не е длъжен да дава публичен достъп
- В абстрактните класове, може да има методи, които не са абстрактни (имат тяло), докато при интерфейсите всички методи са абстрактни.
- Един клас може да имплементира множество интерфейси, докато броят на класовете, които може да наследява е само един.
- Един интерфейс може да наследява един или множество интерфейси.
- Интерфейсите не могат да имат свойства, които да имат различни стойности в отделни инстанции, за разлика от абстрактните класове.

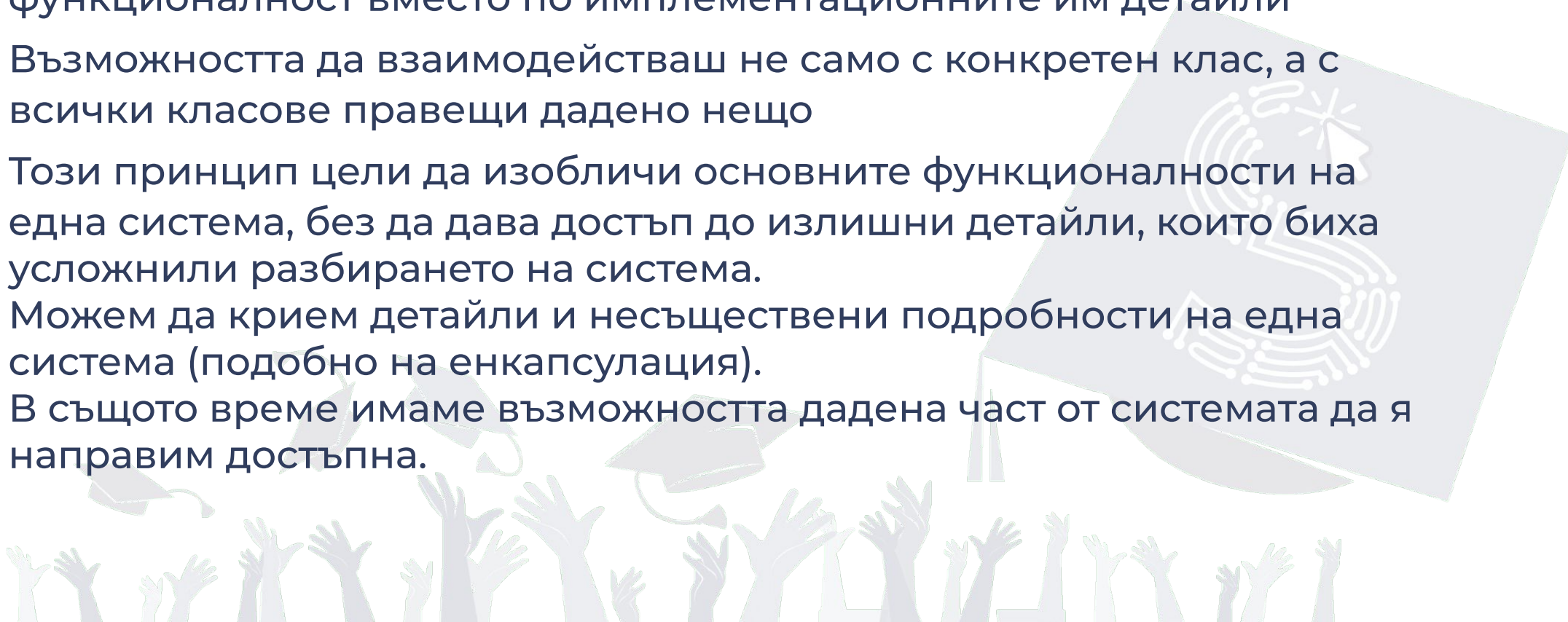
Абстракция



Какво е?



- Създаването на класове, обекти и типове по техните интерфейси и функционалност вместо по имплементационните им детайли
- Възможността да взаимодействаш не само с конкретен клас, а с всички класове правещи дадено нещо
- Този принцип цели да изобличи основните функционалности на една система, без да дава достъп до излишни детайли, които биха усложнили разбирането на система.
- Можем да крием детайли и несъществени подробности на една система (подобно на енкапсулация).
- В същото време имаме възможността дадена част от системата да я направим достъпна.



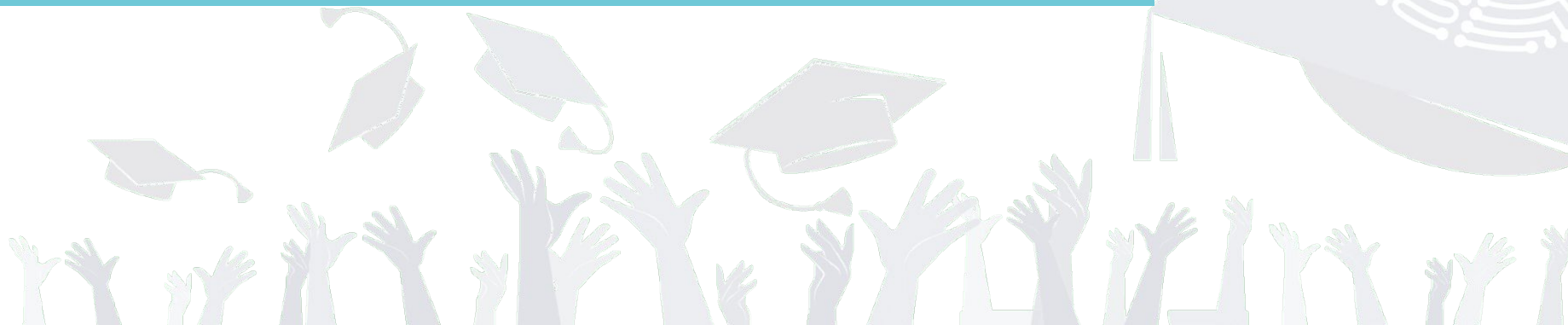
ВЪПРОСИ?



Резюме



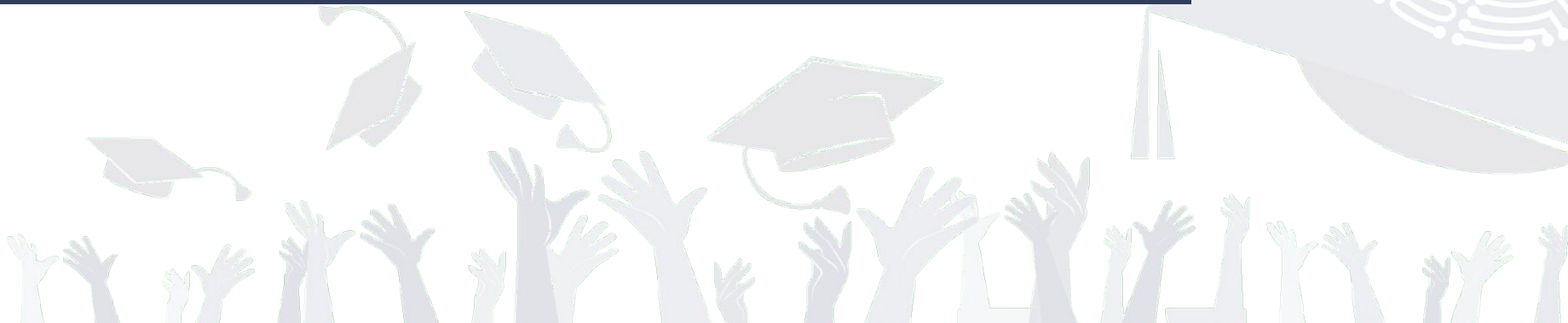
- Абстракцията се постига чрез интерфейси и абстрактни класове.
- Тя ни помага да опростим на пръв поглед системата ни, като скриваме детайлите ѝ.



Ресурси



- [Docs](#)
- [GitHub Repo with Demos](#)
-



Почивка

до 19:45



Задачи за упражняване



Задача 1

- Често големите компании използват т.нар. ERP системи (Enterprise Resource Planning) за да следят клиенто-потока, приходи и разходи, текущи проекти и изработеното време по тези проекти. За целта, създайте програма с два типа потребители - админ и служител. Аминът ще администрира данните в програмата, а служителят ще може да указва колко време по кой проект е работил.



Ами сега?

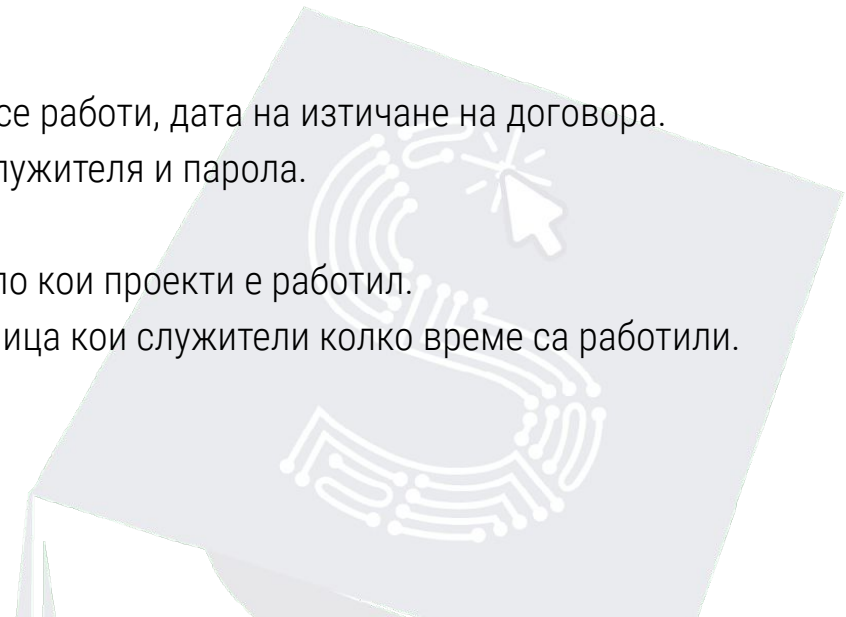
Задача за упражнение

Задача 1



Приложението трябва да:

- Има логин страница
- Има един админ потребител
- При логване с админ потребител, той вижда меню с опции
- Когато се логне админ потребител той може
 - Да въвежда нови клиенти. Всеки клиент има име, име на проекта по който се работи, дата на изтичане на договора.
 - Да регистрира нови служители. Въвежда уникално потребителско име за служителя и парола.
 - Да гледа статистика за служителите
 - Може да търси по име на служител. Така намира той за колко време по кои проекти е работил.
 - Може да търси по номер на седмица. Така вижда за въпросната седмица кои служители колко време са работили.
- Когато се логне служител той може:
 - Да създава протокол за деня в който да пише
 - За кои клиенти е работил днес
 - По колко време на клиент



Ами сега?

Задача за упражнение

Домашно

Качвайте домашното си в ГитХъб и
слагайте линка тук:

<https://forms.gle/AcvCptCbSDizr2Ay6>



Задача 1

Шахът е древна и все още популярна игра. Реализирайте програма, която позволява на играч да играе шах срещу компютъра.

- Играта НЕ изисква графичен интерфейс.
- Играта може да се рисува в конзолата, а играчът да указва фигурата и хода ѝ чрез клавиатурата.
- Трябва да се спазват всички правила на играта шах.
- Играта трябва да има два режима.
- Първият режим е ЛЕСЕН. Компютърът мести фигурите на случаен принцип (разбира се, спрямо правилата).
- Вторият режим е ТРУДЕН. Компютърът има стратегия, която се променя динамично спрямо фигурите на дъската.



© 2020 Нет Ит

БЛАГОДАРЯ ЗА ВНИМАНИЕТО!



SOFTWARE
ACADEMY

