

# Приложно Обектно Ориентирано Програмиране I

Едносеместриален задължителен курс (3+2) за студенти от ИС  
на ФМИ

## Конспект

**Преподавател:** проф. д-р Евгений Кръстев,

**e-mail:** [eck@fmi.uni-sofia.bg](mailto:eck@fmi.uni-sofia.bg)

**Хорариум:** 3 лекции + 2 практически занятия

### **Специалност и курс:**

студенти от 2 и по-горен курс на Информационни системи, Софтуерно инженерство,  
Компютърни науки и Приложна математика

## Анотация

Курсът дава практически умения за прилагане на концепциите на ООП съобразно синтаксиса, техниките и стил за програмиране, използвани в JDK 17+ при решаване на широк кръг информационни проблеми. Тематиката на курса позволява да се усвоят съвременни технологии и добри практики за програмиране Java. Разглеждат се примери, използващи текущата версия на Java, за решаване на широк кръг информационни проблеми. Курсът е включва следните теми: Основи на ООА и моделиране с UML диаграми на класове и дейности; въведение в програмиране със структури от данни на Java, моделиране на интерактивен графичен интерфейс с JavaFX, изграждане на класове с композиция и наследственост, създаване и използване на модули, потребителски пакети от класове, полиморфизъм- абстрактни класове и методи, интерфейси, стил и ефективност при програмиране на Java с приложение на S.O.L.I.D. принципите в ООП; програмна реализация на най-често използвани шаблони в софтуерното инженерство и др.. По време на курса, принципите на ООП се представят посредством пълни и реалистични програмни приложения на Java. Курсът използва IntelliJ като среда за програмиране и се съпровожда с подробни Power Point лекции за представяне на разглеждани теми, от множество примерни програмни решения, упражнения и курсови проекти, които позволяват да се изясни решаването на често срещани бизнес проблеми в предвидените практически занятия.

## Провеждане на курса и изисквания към студентите

Курсът се предлага през **Летния семестър** и е предназначен студенти 2 курс от бакалавърските програми на специалност Информационни системи и Информатика, които имат желание да придобият знания и умения за приложно програмиране на Java. Общи познания за работа с **MS Windows**, начален опит при програмиране на C/C++ са **препоръчителни** изисквания за този курс. Допуска се също, че студентите имат **задоволителни знания по английски език**, позволяващи им да работят и ползват литература в областта на информатиката.

### **Конспект- основни теми**

<b>1.</b>	Въведение в ООП с Java. Типове данни, класове и създаване на обекти. Обектно ориентиран анализ. Основни релации между класове. Програмиране на Java в интегрирана среда за разработка на софтуерни проекти.
<b>2.</b>	Моделиране на класове с UML диаграми. Видове конструктори. Свойства на класове. Моделиране на класове и генериране на съответен сорс код с графичен дизайнер.
<b>3.</b>	Структури за управление на логиката с Java. Моделиране на алгоритми с UML диаграми на дейности. Пресмятания с аритметични и логически изрази.
<b>4.</b>	Видове алгоритми за управление на цикъл. Цикъл, управляван с брояч и цикъл, управляван от ограничение. UML моделиране на алгоритмите и софтуерни приложения.
<b>5.</b>	Реализация на методи на класове в Java. Статични и нестатични методи и данни. Приложения на стандартни пакети от класове за работа математически функции. Приложения, използващи генериране на случайни числа.
<b>6.</b>	Въведение в Java FXML. Предимства на JavaFX. Структура на Java FXML приложение. Основни понятия. Моделиране на потребителски интерфейс с графичен визуален редактор.
<b>7.</b>	Създаване на интерактивен потребителски интерфейс с графичен визуален редактор. Въведение в обработката на събития в графичния потребителски интерфейс.
<b>8.</b>	Масиви- едномерни и многомерни. Синтаксис и основни приложения.
<b>9.</b>	Рекурсия и приложения при сортиране. Метод за сортиране със смесване(merge sort). Метод за бързо сортиране(quick sort). Сложност на алгоритмите.
<b>10.</b>	Моделиране на класове с композиция. Пакетен достъп до класове, методи и данни. Създаване на потребителски пакети от класове
<b>11.</b>	Модулно програмиране на Java. Модулна структура и команди за описание на модулите. Моделиране на класове с наследственост. Софтуерна реализация. UML диаграми на класове в релация на наследственост. Създаване на многократно използвани в Scene Builder визуални контроли
<b>12.</b>	UML диаграми на класове в релация на композиция. Принцип на единичната отговорност. Моделиране на класове с делегиране.
<b>13.</b>	Операции със стрингове, регулярни изрази. Приложения.
<b>14.</b>	Създаване на регистрационни форми с графичен визуален редактор и валидиране на потребителския вход в JavaFX Операции със стрингове, регулярни изрази и приложенията им.
<b>15.</b>	Моделиране на класове с наследственост. Софтуерна реализация. UML диаграми на класове в релация на наследственост.

<b>16.</b>	Полиморфизъм- абстрактни класове и интерфейси. Приложения на полиморфизъм
<b>17.</b>	Принцип на Liskov и приложенията му.
<b>18.</b>	Побитови операции. Приложения. Създаване на “маска”. Пресмятане на разстояние по Hamming Криптиране на данни.
<b>19.</b>	Вътрешни и анонимни класове. Closure и Callback софтуерни конструкции. Създаване на потребителски събития. Йерархия на наследственост на класовете производни на class.
<b>20.</b>	Обработка на изключения в Java. Синтаксис на блока за прихващане и обработка на изключенията. Throwable. Проверявани и непроверявани изключения. Дефиниране на потребителски типове изключения
<b>21.</b>	Параметричен полиморфизъм (generics). Шаблони на методи, класове и интерфейси с параметри за тип. Реализация на параметризираните по тип шаблони в Java. Ограничения в използването на параметрите за тип. Наследственост с използване на параметри за тип.
<b>22.</b>	Ковариантност, контравариантност и инвариантност на типове данни, използващи параметри за тип. Приложение на Принципа GET- PUT. Ограничения отгоре и ограничения отдолу за стойностите на параметрите за тип.
<b>23.</b>	Колекция от структури на данни в Java. Имплементации на интерфейси Collection и Map. Задаване на критерии за сравнение с имплементиране на интерфейс Comparator.
<b>24.</b>	Ламбда изрази и Функционални интерфейси. Стандартни Функционални интерфейси. Статични, подразбиращи се и private методи в интерфейс..
<b>25.</b>	Stream приложен програмен интерфейс в Java. Създаване и изпълнение на верига от команди. Междинни и завършващи операции. Търсене, филтриране, сортиране и групиране на данни. Производни на Stream<T> интерфейси за работа с примитивни данни. Приложения при обработка на структури от данни.
<b>26.</b>	Обработка на събития породени от движение на мишка. Приложение за свободно рисуване в JavaFX. Обработка на събития, породени от натискане на клавиш от клавиатурата
<b>27.</b>	JavaFX свойства, създаване на JavaFX свойства, видове свързване на свойства и приложения. Обработка на събитието Change на обекти от тип ObservableValue.
<b>28.</b>	Шаблони в софтуерното инженерство. Класификация на шаблоните за моделиране. Софтуерна реализация на Java на Observer шаблон. JavaFX Observable и производни на него ObservableList и ObservableMap. Lazy/Eager evaluation. Обработка на събитията Change и Invalidation.
<b>29.</b>	Статични методи на FXCollections. Връзка с Java Collections. ListChangeListener и MapChangeListener. ComboBox и приложенията му в реализацията JavaFX приложения. Примерна реализация на потребителски интерфейс за Кошница за пазаруване с ComboBox и Binding

	на JavaFX свойства.
<b>30.</b>	Създаване на TableView control, конфигуриране на TableView и приложения на TableView с SQLite база от данни. Създаване на Connection към база от данни и изпълнение на SQL SELECT заявка с Java.

**Забележка:** *Задачите* представени като част от упражненията за илюстрация на лекциите са **неделима част от този конспект** и техните решения ще **се изискват в края на**.

### Изпити и оценки

Окончателната оценка се базира на **писмен изпит** (40%) по *средата на курса*, **финален писмен изпит** (40%) и изпълнение на **домашни и кратки контролни** (20%).

**Писменият изпит и защитата на курсовия проект не се повтарят след приключване на семестъра.**

Оценките се формират по следната таблица:

<b>2</b>	<b>от 0 до 54 точки</b>
<b>3</b>	<b>от 55 до 64 точки</b>
<b>4</b>	<b>от 65 до 74 точки</b>
<b>5</b>	<b>от 75 до 84 точки</b>
<b>6</b>	<b>от 85 до 100 точки</b>

### Литература

[1] **P. Deitel, H. Deitel**, "Java How to Program (Early objects)", Prentice Hall 11th ed. 2018, ISBN-13: 9780134751863 (**основна**)

[2] **Y. Daniel Liang**, "Java Programming. Comprehensive version", **12<sup>th</sup>** ed., Prentice Hall **2020** ISBN-13: 978-0-13-651996-6

[3] **Bruce Eckel** "Thinking in Java", 4th ed., Prentice Hall 2006 или българското ѝ издание "Да мислим на Java" том 1 и 2, SoftPress, **2001**

януари, 2022