

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Львівський національний університет імені Івана Франка
Факультет прикладної математики та інформатики
Кафедра інформаційних систем

Затверджено

На засіданні кафедри інформаційних систем
факультету прикладної математики та
інформатики
Львівського національного університету
імені Івана Франка
(протокол № 1 від 28 серпня 2023 р.)

Завідувач кафедри Шинкаренко Г.А.



Силабус з навчальної дисципліни
«Програмування на Java»,
що викладається в межах ОПП Середня освіта (Інформатика)
першого (бакалаврського) рівня вищої освіти
для здобувачів з спеціальності 014.09 середня освіта
(Інформатика)

Львів 2023 р.

Назва дисципліни	Програмування на Java
Адреса викладання дисципліни	Головний корпус ЛНУ ім. І. Франка м. Львів, вул. Університетська 1
Факультет та кафедра, за якою закріплена дисципліна	Факультет прикладної математики та інформатики Кафедра інформаційних систем
Галузь знань, шифр та назва спеціальності	Галузь знань: 01 Освіта / Педагогіка Спеціальність: 014 Середня освіта Спеціалізація: 014.09 Середня освіта (Інформатика)
Викладачі дисципліни	Бернакевич Ірина Євстахіївна, к. ф.-м. н., доцент кафедри інформаційних систем;
Контактна інформація викладачів	Електронна пошта: iryna.bernakevych@lnu.edu.ua ; веб-сторінка: https://ami.lnu.edu.ua/employee/bernakevych ; Головний корпус ЛНУ ім. І. Франка, каб. 261. м. Львів, вул. Університетська, 1
Консультації з питань навчання по дисципліні відбуваються	Консультації проводять раз на тиждень згідно з оприлюдненим розкладом консультацій викладача. Можливі онлайн консультації через Microsoft Teams.
Сторінка курсу	https://ami.lnu.edu.ua/course/prohramuvannia-movoju-java-so
Інформація про дисципліну	Курс «Програмування на Java» є дисципліною вільного вибору студента зі спеціальності 014 Середня освіта для освітньо-професійної програми «Середня освіта (Інформатика)», яку викладають в п'ятому семестрі в обсязі 5-ох кредитів (за Європейською Кредитно-Трансферною Системою ECTS).
Коротка анотація дисципліни	Курс розроблено таким чином, щоб навчити студентів об'єктно-орієнтованого програмування з використанням мови програмування Java. Основний акцент робиться на вивчення можливостей бібліотеки JDK, а саме колекцій та узагальнень, можливостей побудови багатопотокових застосунків, роботи із графікою. Розглянуто можливості створення графічного інтерфейсу користувача з використанням бібліотек AWT і Swing. Основну частину курсу займає розгляд практичних аспектів побудови багатопотокових застосунків та застосунків для взаємодії з базами даних.
Мета та цілі дисципліни	Метою курсу «Програмування на Java» є освоєння студентами теоретичних і практичних основ програмування мовою Java та принципів розробки програмного забезпечення з використанням Java. Ціллю курсу є досягнення достатнього рівня знань та практичних навичок для створення Java застосунків на належному для сучасних вимог рівні.
Література для вивчення дисципліни	<i>Основна література</i> 1. Васильєв О. Програмування мовою Java. – Тернопіль: Навчальна книга – Богдан, 2020. – 696 с. 2. Кеті Сьєрра, Берт Бейтс Head First. Java. Легкий для сприйняття довідник. – Харків: Фабула, 2022. – 720 с. 3. Галкін О.В., Катеринич Л.О., Шкільняк О.С. Програмування на Java 8: Навчальний посібник для студентів факультету комп'ютерних наук та кібернетики. – К.: ЛОГОС, 2017. – 186 с. 4. Кадомський К.К., Ніколюк П.К. Java. Теорія і практика. – Вінниця, 2018.- 198 с.

	<p>5. Еванс Б., Кларк Д., Вербург М. The Well-Grounded Java Developer, Manning, 2023 – 704 с.</p> <p>6. Тарнавський Ю. А. Java-програмування: комп'ютерний практикум [Електронний ресурс]: навч. посіб. для студ. спеціальності 122 «Комп'ютерні науки». – Київ: КПІ ім. Ігоря Сікорського, 2021. – 95 с. <i>Допоміжна література</i></p> <p>7. Horstmann C. S. Core Java, Volume I: Fundamentals, 12th Edition, Addison-Wesley Professional, 2021.</p> <p>8. Horstmann C. S. Core Java, Vol. II-Advanced Features, 12th Edition, Oracle Press, 2022.</p> <p>9. Schildt H. Java: The Complete Reference, 12th Edition, McGraw-Hill, 2021.</p> <p>10. Deitel P., Deitel H. Java How to Program, Early Objects: 11th Edition, Pearson, 2017, 1296 p.</p> <p>11. Deitel P., Deitel H. Java How To Program, Late Objects: 11th Edition, Pearson, 2017, 1248 p.</p>
Обсяг курсу	Загальний обсяг: 150 годин. Аудиторних занять: 64 год., з них 32 год. лекцій та 32 години лабораторних робіт. Самостійної роботи: 86 год.
Очікувані результати навчання	<p>Після завершення цього курсу студент буде <i>знати</i>:</p> <ul style="list-style-type: none"> – основні конструкції мови програмування Java, – принципи побудови об'єктно-орієнтованих програм, – структуру колекцій та узагальнення; – принципи побудови багатопотокових програм; – модель обробки подій; – бібліотеки для створення графічних застосунків; – засоби роботи з базами даних. <p><i>вміти</i>:</p> <ul style="list-style-type: none"> – створювати класи та інтерфейси, ієрархії класів; – використовувати узагальнення та можливості останніх версій java ; – розробляти багатопоточні програми; – створювати програми з привабливим графічним інтерфейсом; – встановлювати взаємодію застосунків з базами даних.
Компетентності	<p><i>Загальні (ЗК):</i></p> <p>ЗК1. Здатність до абстрактного мислення, аналізу та синтезу.</p> <p>ЗК2. Здатність вчитися й оволодівати сучасними знаннями. Застосовувати знання у практичних ситуаціях</p> <p>ЗК3. Знання та розуміння предметної області та розуміння професійної діяльності.</p> <p>ЗК6. Здатність до застосування сучасних інформаційних і комунікаційних технологій у освітній діяльності</p> <p>ЗК7. Здатність генерувати нові ідеї (креативність).</p> <p><i>Спеціальні (фахові, предметні) компетентності (СК):</i></p> <p>СК7. Здатність до логічного мислення, побудови логічних висновків, використання формальних мов і моделей алгоритмічних обчислень, проектування, розроблення й аналізу алгоритмів, оцінювання їх ефективності та складності, розв'язності та нерозв'язності алгоритмічних проблем для адекватного моделювання предметних областей і створення програмних та інформаційних систем.</p> <p>СК11. Здатність проектувати та розробляти програмне забезпечення із застосуванням різних парадигм програмування: узагальненого, об'єктно-орієнтованого, функціонального, логічного, з відповідними моделями, методами й алгоритмами обчислень, структурами даних і механізмами управління.</p> <p>СК13. Здатність застосовувати методології, технології та інструментальні засоби для управління процесами життєвого циклу інформаційних і програмних систем,</p>

	продуктів і сервісів інформаційних технологій відповідно до вимог замовника.
Програмні результати навчання	<p>ПР8. Застосовувати знання основних форм і законів абстрактно-логічного мислення, основ методології наукового пізнання, форм і методів вилучення, аналізу, обробки та синтезу інформації в предметній області комп'ютерних наук.</p> <p>ПР11. Проектувати, розробляти та аналізувати алгоритми розв'язання обчислювальних та логічних задач, оцінювати ефективність та складність алгоритмів на основі застосування формальних моделей алгоритмів та обчислюваних функцій.</p> <p>ПР14. Розробляти програмні моделі предметних середовищ, вибирати парадигму програмування з позицій зручності та якості застосування для реалізації методів та алгоритмів розв'язання задач в галузі комп'ютерних наук.</p> <p>ПР16. Застосовувати знання методології та CASE-засобів проектування складних систем, методів структурного аналізу систем, об'єктно-орієнтованої методології проектування при розробці і дослідженні функціональних моделей організаційно-економічних і виробничо-технічних систем.</p>
Ключові слова	<p>Типи даних, оператори, об'єктно-орієнтований підхід, класи, інтерфейси, пакети, конструктори, специфікатори доступу, внутрішні класи, анонімні класи, ієрархії класів, виняткові ситуації, колекції, ітератори, узагальнення, багатопоточність, об'єкти синхронізації, графічний контекст пристрою, модель обробки подій, класи подій, слухачі подій, AWT, Swing, менеджери розміщення, архітектура MVC, модель компонента, візуалізація компонента, обробка подій розширених компонентів, архітектура JDBC, класифікація драйверів JDBC, URL бази даних, інструкції SQL, набори результатів, метадані бази даних, метадані набору результатів, оновлювані набори даних, пакетні зміни в базі даних, збережені процедури, розподілені транзакції, управління транзакціями, режими транзакцій, рівні ізоляції транзакцій.</p>
Формат курсу	<p>Очний: проведення лекцій, лабораторних робіт та консультацій в приміщеннях університету, а в умовах карантину – онлайнний на платформі Microsoft Teams</p>
Теми	<ol style="list-style-type: none"> Синтаксис мови Java. Основні лексеми мови Java. Огляд типів даних. Оператори. Об'єктно-орієнтований підхід до створення Java-програм. Оголошення класів. Специфікатори доступу. Використання ключових слів <code>this</code> та <code>super</code>. Конструктори та метод <code>finalize()</code>. Внутрішні класи (класи-члени, локальні класи, анонімні класи). Похідні класи та динамічна диспетчеризація методів. Інтерфейси та їх використання. Особливості створення пакетів. Обробка виняткових ситуацій. Типи виняткових ситуацій. Генерація та перехоплення винятків. Вбудовані виняткові ситуації. Створення власних класів виняткових ситуацій. Зберігання даних (колекції, алгоритми). Структура колекцій. Інтерфейси колекцій. Класи колекцій. Доступ до елементів колекцій через ітератор. Робота з картами. Компаратори. Основні алгоритми колекцій. Зберігання даних (узагальнення). Узагальнення. Використання шаблонних аргументів. Узагальнені класи, інтерфейси, методи. Ієрархії узагальнених типів. Обмеження шаблонів. Багатопотокове програмування та утиліти паралельного програмування. Модель потоку Java і його властивості. Створення потоку. Тіло потоку та його життєвий цикл. Пріоритети потоків. Потоки-демони. Синхронізація та взаємне блокування. Використання об'єктів синхронізації (семафорів, циклічних бар'єрів, та ін.). Паралельні колекції. Блокувальники.

	<p>7. Потоки введення-виведення. Стандартні системні потоки введення-виведення (java.System.lang). Робота з файлами і каталогами. Ієрархія класів байтового (символьного) введення. Ієрархія класів байтового (символьного) виведення. Серіалізація об'єктів.</p> <p>8. Робота з графікою. Graphics 2D. Графічний контекст пристрою. Використання примітивної графіки. Робота зі шрифтами. Загальні принципи рисування з використанням Graphics 2D. Інтерфейс Shape. Рисування прозорих форм. Стили рисування. Використання локальних шрифтів. Створення пера та його характеристики. Перетворення координат (зсув, масштабування, поворот).</p> <p>9. Обробка подій Java. Модель обробки подій в Java. Класи подій. Слухачі подій, їх методи та інтерфейси. Класи-адаптери.</p> <p>10. Програмування інтерфейсу користувача з використанням AWT. Стандартні компоненти та контейнери. Менеджери розміщення. Створення вікна застосування. Побудова меню. Діалогові вікна. Використання діалогових вікон для доступу до локальної файлової системи. Елементи пакету AWT та породжувані ними події.</p> <p>11. Програмування інтерфейсу користувача з використанням Swing. Особливості Swing-компонентів. Архітектура MVC. Типи контейнерів. Прості компоненти. Іконки, підказки, рамки. Лінійні регулятори. Текстові компоненти. Побудова меню. Створення панелі інструментів.</p> <p>12. Розширені компоненти Swing. Списки, дерева, таблиці та обробка подій. Використання моделі компонентів. Візуалізація компонентів.</p> <p>13. Основи JDBC. Архітектура JDBC. Класифікація драйверів JDBC. Конфігурація JDBC-драйвера. З'єднання з БД та налаштування його властивостей. URL бази даних. Передача інструкцій SQL. Попередньо відкомпільовані інструкції. Набори результатів та їх обробка. Обробка множинних результатів запиту. Отримання відомостей про набори результатів. Відповідність між типами SQL та Java. Метадані бази даних і та їх використання Управління помилками та попередженнями.</p> <p>14. Розширені засоби JDBC. Набори результатів із розширеними засобами перегляду. Оновлювані набори даних. Пакетні зміни в базі даних. Розширені типи даних. Набори записів, їх реалізація та управління ними. Збережені процедури, їх створення та виконання. Параметри IN та OUT. Іменування бази даних за допомогою JNDI. Розподілені транзакції. Управління транзакціями. Режими транзакцій та рівні ізоляції транзакцій.</p>
Підсумковий контроль, форма	Залік у кінці семестру
Пререквізити	<p>Для вивчення курсу студенти потребують базових знань з</p> <ul style="list-style-type: none"> – Програмування; – Алгоритми та структури даних;
Навчальні методи та техніки, які будуть використовуватися під час викладання курсу	Презентації, лекції, виконання та оцінювання індивідуальних завдань, самостійна робота з вивченням оприлюднених електронних матеріалів. Проведення тестування студентів на платформі e-learning.lnu.edu.ua.
Необхідне обладнання	Для проведення лекцій: комп'ютер, проектор, доступ до мережі Інтернет. Для проведення лабораторних та виконання завдань: комп'ютер, ОС Windows/Linux, доступ до Інтернету, середовище програмування мовою Java (Intrllij IDEA, Eclipse IDE тощо).

<p>Критерії оцінювання (окремо для кожного виду навчальної діяльності)</p>	<p>Оцінювання проводиться за 100-бальною шкалою. Бали нараховуються за наступним співвідношенням:</p> <ul style="list-style-type: none"> – індивідуальні завдання: 50% семестрової оцінки; максимальна кількість балів 50 – залік: 50% семестрової оцінки; максимальна кількість балів 50 <p>Підсумкова максимальна кількість балів 100.</p> <p>Академічна доброчесність: Очікується, що роботи студентів будуть їх оригінальними дослідженнями чи міркуваннями. Відсутність посилань на використані джерела, фабрикування джерел, списування, втручання в роботу інших студентів становлять, але не обмежують, приклади можливої академічної недоброчесності. Виявлення ознак академічної недоброчесності в письмовій роботі студента є підставою для її незарахування викладачем, незалежно від масштабів плагіату чи обману.</p> <p>Відвідання занять є важливою складовою навчання. Очікується, що всі студенти відвідають усі лекції та практичні заняття курсу. Студенти повинні інформувати викладача про неможливість відвідати заняття. У будь-якому випадку студенти зобов'язані дотримуватися термінів визначених для виконання всіх видів письмових робіт та індивідуальних завдань, передбачених курсом.</p> <p>Література. Уся література, яку студенти не зможуть знайти самостійно, буде надана викладачем виключно в освітніх цілях без права її передачі третім особам. Студенти заохочуються до використання також й іншої літератури та джерел, яких немає серед рекомендованих.</p> <p>Політика виставлення балів. Враховуються бали набрані при поточному тестуванні, самостійній роботі та бали підсумкового тестування. При цьому обов'язково враховуються присутність на заняттях та активність студента під час практичного заняття; недопустимість пропусків та запізнень на заняття; користування мобільним телефоном, планшетом чи іншими мобільними пристроями під час заняття в цілях не пов'язаних з навчанням; списування та плагіат; несвоєчасне виконання поставленого завдання і т. ін.</p> <p>Жодні форми порушення академічної доброчесності не толеруються.</p>
<p>Питання до заліку чи екзамену.</p>	<ol style="list-style-type: none"> 1. Оголошення класів, інтерфейсів, пакетів. Специфікатори доступу 2. Використання ключових слів this та super 3. Вкладені класи, внутрішні класи, анонімні класи 4. Створення ієрархій класів 5. Обробка виняткових ситуацій. Контрольовані та неконтрольовані ВС 6. Колекції, узагальнення 7. Багатопотокові програми. Синхронізація та взаємне блокування. Використання об'єктів синхронізації 8. Потоки введення-виведення. Файлове введення-виведення. Серіалізація об'єктів. 9. Загальні принципи рисування з використанням Graphics 2D. 10. Інтерфейс Shape. Стили рисування. Створення пера та його характеристики 11. Перетворення координат (зсув, масштабування, поворот) 12. Модель обробки подій в Java. Класи подій. Слухачі подій, їх методи та інтерфейси. 13. Стандартні компоненти та контейнери AWT 14. Менеджери розміщення 15. Побудова меню. Діалогові вікна 16. Особливості Swing-компонентів. Архітектура MVC 17. Створення панелі інструментів 18. Розширені компоненти Swing. Обробка подій розширених

	<p>компонентів</p> <ol style="list-style-type: none">19. Архітектура JDBC. Класифікація драйверів JDBC20. Передача інструкцій SQL. Попередньо відкомпільовані інструкції21. Набори результатів та їх обробка22. Метадані бази даних та наборів результатів23. Режими транзакцій та рівні ізоляції транзакцій
Опитування	Анкету-оцінку з метою оцінювання якості курсу буде надано по завершенню курсу.