

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 29.08 2025 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 А4.09 – Середня освіта (Інформатика)

Львів 2025 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 | A – Освіта/Педагогіка A4.09 – Середня освіта (Інформатика) |
| Викладачі дисципліни | Бернакевич Ірина Євстахіївна, доцент кафедри інформаційних систем |
| Контактна інформація викладачів | iryna.bernakevych@lnu.edu.ua ; 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 в день проведення лекцій/практичних занять (за попередньою домовленістю). |
| Сторінка курсу | https://ami.lnu.edu.ua/course/prohramuvannia-movoju-java-so |
| Інформація про дисципліну | Дисципліна “Програмування на Java” є дисципліною вільного вибору студента з спеціальності A4.09 – Середня освіта (Інформатика) для освітньої програми Середня освіта (Інформатика), яка викладається в 5-му семестрі в обсязі 4-ох кредитів (за Європейською Кредитно-Трансферною Системою ECTS). |
| Коротка анотація дисципліни | Курс розроблено таким чином, щоб надати учасникам знання принципів об’єктно-орієнтованого програмування з використанням 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 вивчення дисципліни вільного вибору “Програмування на Java” є освоєння студентами теоретичних і практичних основ програмування мовою Java та принципів розробки програмного забезпечення з використанням Java. |
| Література для вивчення дисципліни | <ol style="list-style-type: none"> 1. Herbert Schildt. Java: The Complete Reference, 12th Edition, 2021. – 1280 p. 2. Cay S. Horstmann and Gary Cornell. Core Java volume I: fundamentals, 9th edition. 2014.– 832 p. 3. Cay S. Horstmann and Gary Cornell. Core Java 2. 2015.– 864 p. 4. X.Y. Wang Java Spring Framework: 100 Interview Questions. Kindle Edition 2018, 563p. 5. X.Y. Wang Design Patterns in Java: Interview Questions and Answers . Kindle Edition 2019, 687p. 6. https://coursera.org/projects/object-oriented-programming-java |
| Обсяг курсу | Загальний обсяг: 120 годин. Аудиторних занять: 64 год., з них 32 год. лекцій та 32 години лабораторних робіт. Самостійної роботи: 56 год. |

| | |
|---|---|
| <p>Очікувані результати навчання</p> | <p>Після завершення цього курсу студент буде знати:</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>Вміти:</p> <ul style="list-style-type: none"> - створювати класи та інтерфейси, ієрархії класів; - використовувати узагальнення та нові можливості java 8; - розробляти багатопоточні програми; - створювати програми з привабливим графічним інтерфейсом; - встановлювати взаємодію застосунків з базами даних. |
| <p>Ключові слова</p> | <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, багато поточність, об'єкти синхронізації, графічний контекст пристрою. примітивна графіка, стилі рисування, 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> |
| <p>Формат курсу</p> | <p>Очний, дистанційний Проведення лекцій, лабораторних робіт і консультацій. Ознайомлення з Internet курсами по Java Open University courses https://www.openuniversity.edu/courses/qualifications/details/m250?orig=q62-soft або COURSERA courses: https://ru.coursera.org/specializations/core-java https://ru.coursera.org/projects/object-oriented-programming-java</p> |

| Теми | <ol style="list-style-type: none"> <li data-bbox="459 120 1513 450">1. Синтаксис мови Java та об'єктно-орієнтований підхід до створення Java-програм. Основні лексеми мови Java. Огляд типів даних. Оператори. 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. <li data-bbox="459 456 1513 674">2. Зберігання даних (колекції, алгоритми, узагальнення). Структура колекцій. Інтерфейси колекцій. Класи колекцій. Доступ до елементів колекцій через ітератор. Робота з картами. Компаратори. Основні алгоритми колекцій. Узагальнення. Використання шаблонних аргументів. Узагальнені класи, інтерфейси, методи. Ієрархії узагальнених типів. Обмеження шаблонів. <li data-bbox="459 680 1513 898">3. 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. <li data-bbox="459 904 1513 1189">4. Робота з графікою. Graphics 2D. Обробка подій Java. Графічний контекст пристрою. Використання примітивної графіки. Робота зі шрифтами. Загальні принципи рисування з використанням Graphics 2D. Інтерфейс <code>Shape</code>. Рисування прозорих форм. Стили рисування. Використання локальних шрифтів. Створення пера та його характеристики. Перетворення координат (зсув, масштабування, поворот). Модель обробки подій в Java. Класи подій. Слухачі подій, їх методи та інтерфейси. <li data-bbox="459 1196 1513 1368">5. 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. <li data-bbox="459 1375 1513 1592">6. 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. Розширені компоненти Swing (списки, дерева, таблиці) та обробка подій. Використання моделі компонентів. Візуалізація компонентів. <li data-bbox="459 1599 1513 1861">7. 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. <li data-bbox="459 1868 1513 2107">8. 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. |
|------|--|
|------|--|

| | |
|---|---|
| Підсумковий контроль, форма | Залік у кінці семестру |
| Пререквізити | Для вивчення курсу студенти потребують базових знань з <ul style="list-style-type: none"> - Програмування; - Алгоритми та структури даних; |
| Навчальні методи та техніки, які будуть використовуватися під час викладання курсу | Презентації, лекції Індивідуальні завдання |
| Необхідне обладнання | Комп'ютер із програмним забезпеченням JDK, інтегрованим середовищем розробки IntrlIj IDEA або Eclipse IDE |
| Критерії оцінювання (окремо для кожного виду навчальної діяльності) | <p>Оцінювання проводиться за 100-бальною шкалою. Бали нараховуються за наступним співвідношенням: 60 балів нараховують за виконання лабораторних завдань: по 10 балів за кожну з 6 залікових лабораторних робіт. Ще 40 балів – за створення індивідуального завдання.</p> <p>Критерії оцінювання лабораторних та індивідуальних завдань: 100% балів – студент повністю виконав умови завдання, хід розв'язування оформлено належним чином, нема ознак недоброчесності. 80% балів – студент повністю виконав умови завдання, хід розв'язування оформлено належним чином, розв'язок містить незначні помилки, нема ознак недоброчесності. 60-50% балів – студент виконав завдання з помилками, або ж не повністю викладено хід розв'язування завдання, нема ознак недоброчесності; 20% балів – студент виконав завдання частково з грубими помилками, які самостійно не може виправити, демонструє незнання матеріалу, нема ознак недоброчесності. 0 балів – студент не виконав завдання або ж у роботі виявлені ознаки недоброчесності.</p> <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 зайняття 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</p> |

| | |
|---------------------------------------|--|
| | <p>обов'язково враховуються присутність на заняттях та активність студента під час практичного заняття; недопустимість пропусків та запізнь 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> |
| Питання до заліку чи екзамену. | <p>Оголошення класів, інтерфейсів, пакетів. Специфікатори доступу</p> <p>Використання ключових слів <code>this</code> та <code>super</code></p> <p>Вкладені класи, внутрішні класи, анонімні класи</p> <p>Створення ієрархій класів</p> <p>Обробка виняткових ситуацій. Контрольовані та неконтрольовані ВС</p> <p>Колекції, узагальнення</p> <p>Багатопотокові програми. Синхронізація та взаємне блокування.</p> <p>Використання об'єктів синхронізації</p> <p>Загальні принципи рисування з використанням Graphics 2D</p> <p>Інтерфейс Shape. Стилі рисування. Створення пера та його характеристики</p> <p>Перетворення координат (зсув, масштабування, поворот)</p> <p>Модель обробки подій в Java. Класи подій. Слухачі подій, їх методи та інтерфейси.</p> <p>Стандартні компоненти та контейнери AWT</p> <p>Менеджери розміщення</p> <p>Побудова меню. Діалогові вікна</p> <p>Особливості Swing-компонентів. Архітектура MVC</p> <p>Створення панелі інструментів</p> <p>Розширені компоненти Swing. Обробка подій розширених компонентів</p> <p>Архітектура JDBC. Класифікація драйверів JDBC</p> <p>Передача інструкцій SQL. Попередньо відкомпільовані інструкції</p> <p>Набори результатів та їх обробка</p> <p>Метадані бази даних та наборів результатів</p> <p>Режими транзакцій та рівні ізоляції транзакцій</p> |
| Опитування | <p>Анкету-оцінку з метою оцінювання якості курсу буде надано по завершенню курсу.</p> |