

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Львівський національний університет імені Івана Франка
Відокремлений структурний підрозділ
«Педагогічний фаховий коледж
Львівського національного університету імені Івана Франка»

ЗАТВЕРДЖЕНО

На засіданні циклової комісії
спеціальності "Комп'ютерні науки"
протокол № 1 від «30» серпня 2024 р.

Голова циклової комісії

 Олександр КНИГІНЦЬКИЙ

СИЛАБУС З НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
«Програмування мовою Java»

Освітньо-професійний ступінь	«фаховий молодший бакалавр»
Галузь знань	12 Інформаційні технології
Спеціальність	122 Комп'ютерні науки
Освітньо-професійна програма	Комп'ютерні науки
Форма навчання	денна

Назва дисципліни	Програмування мовою Java
Адреса викладання дисципліни	м. Львів, вул. Тарнавського, 107
Факультет та кафедра, за якою закріплена дисципліна	Відокремлений структурний підрозділ «Педагогічний фаховий коледж Львівського національного університету імені Івана Франка». Циклова комісія спеціальності Комп'ютерні науки
Галузь знань, шифр та назва спеціальності	12 Інформаційні технології, 122 Комп'ютерні науки
Викладачі дисципліни	Майорак Богдан Ярославович
Контактна інформація викладачів	maiorchakbohdan@gmail.com
Консультації з питань навчання по дисципліні відбуваються	Консультації проводяться в день проведення лекційних занять (за попередньою домовленістю) у корпусі по вул. Ген. Тарнавського, 107, Ауд.303. В режимі он-лайн: консультації проводяться за домовленістю з студентами чи групою студентів на платформі MS Teams та на інших електронних ресурсах, за попереднім погодженням часу (через електронну пошту викладача або телефонний дзвінок).
Сторінка дисципліни	https://pedcollege.lnu.edu.ua/course/programuvania-na-java/
Інформація про дисципліну	Дисципліна «Програмування мовою Java» є вибірковою освітньою компонентом ОПП циклу професійної та практичної підготовки зі спеціальності 122 Комп'ютерні науки для освітньої програми «Комп'ютерні науки», яка викладається в 4 семестрі в обсязі 3-х кредитів (за Європейською Кредитно-Трансферною Системою ECTS).
Коротка анотація дисципліни	Навчальна дисципліна «Програмування мовою Java» спрямована на формування у студентів знань та вмінь щодо розробки програмного забезпечення з використанням сучасних технологій та інструментів Java. Вона охоплює основи синтаксису мови, принципи об'єктно-орієнтованого програмування, роботу з колекціями, потоками, файлами та базами даних. Особлива увага приділяється використанню стандартних бібліотек, розробці багатопотокових додатків, роботі з мережею, а також створенню веб-додатків із застосуванням Java EE та Spring Framework.
Мета та цілі дисципліни	Мета: Надати студентам глибокі знання та практичні навички з розробки програмного забезпечення мовою Java . Студенти навчатимуться застосовувати об'єктно-орієнтоване програмування, працювати з бібліотеками та фреймворками, створювати консольні, десктопні та веб-додатки, а також розробляти продуктивні та масштабовані програмні рішення. Цілі: <ul style="list-style-type: none"> • Ознайомити студентів із основами мови Java, її синтаксисом, ключовими конструкціями та принципами об'єктно-орієнтованого програмування (ООП).

	<ul style="list-style-type: none"> • Вивчити роботу з колекціями, потоками введення/виведення, багатопотоковістю та роботою з файлами. • Навчити застосовувати патерни проектування для створення ефективних програмних рішень.
Література для вивчення дисципліни	<p>Основна література:</p> <ol style="list-style-type: none"> 1. Сьєрра К., Бейтс Б. <i>Вивчаємо Java</i>. – Київ: Вид-во O'Reilly, 2023. 2. Васильєв О. <i>Програмування мовою Java</i>. – Львів: Вид-во «Новий Світ», 2020. 3. Мартін Р. <i>Чистий код. Створення, аналіз та рефакторинг</i>. – Київ: Вид-во «Фактор», 2019. 4. Еккель Б. <i>Філософія Java</i>. – Київ: Вид-во «Факт», 2018. 5. Грошко М., Коваль В. <i>Основи програмування Java</i>. – Київ: НаУКМА, 2022.
Обсяг курсу	51 години аудиторних занять. З них 17 години лекцій, 34 години практичних робіт на кожну із 8 бригад та 39 годин самостійної роботи
Очікувані результати навчання	<p>В результаті вивчення даного курсу студент повинен:</p> <p>Знати:</p> <ul style="list-style-type: none"> • Основи мови Java, її синтаксис, типи даних, ключові конструкції та принципи об'єктно-орієнтованого програмування (ООП). • Архітектуру Java-платформи, віртуальну машину Java (JVM), механізм збирача сміття та управління пам'яттю. • Основні бібліотеки та фреймворки Java, зокрема Collections API, Streams API, JDBC, JPA, Hibernate, Spring Framework. • Основи багатопотокового програмування та конкурентності у Java. • Роботу з мережею, створення веб-додатків та REST API на Java. • Методи тестування Java-додатків за допомогою JUnit, Mockito та принципи CI/CD для Java-проектів. <p>Вміти:</p> <ul style="list-style-type: none"> • Розробляти консольні, десктопні та веб-додатки мовою Java. • Використовувати об'єктно-орієнтоване програмування (ООП) та патерни проектування у розробці програмного забезпечення. • Працювати з базами даних, застосовуючи JDBC, Hibernate, JPA. • Розробляти веб-сервіси та REST API за допомогою Spring Boot. • Використовувати засоби багатопотокового програмування та конкурентності для підвищення продуктивності програм. • Працювати з засобами тестування, налаштовувати автоматизовані тести та інтегрувати їх у CI/CD-процеси.

Загальні компетентності

ЗК4. Здатність застосовувати знання у практичних ситуаціях.

ЗК7. Здатність спілкуватися іноземною мовою.

ЗК8. Здатність вчитися і оволодівати сучасними знаннями.

Спеціальні (фахові) компетентності

ФК3. Здатність розробляти, аналізувати та застосовувати ефективні алгоритми для розв'язання конкретних професійних задач залежно від предметного середовища.

ФК4. Здатність здійснювати проектування та розробку програмного забезпечення.

**Програмні
результати
навчання**

ФК9. Здатність застосовувати знання сучасних методів і технологій створення та супроводження розподілених систем.

ПРН05. Розуміти основні методи і технології об'єктно-орієнтованого та компонентного програмування.

ПРН10. Знати методології, методи, моделі, процеси і технології життєвого циклу розробки та тестування програмного забезпечення.

ПРН11. Застосовувати сучасні мови програмування та технології для розробки програмного забезпечення розподілених систем.

ПРН12. Знати основні принципи функціонування системного та прикладного програмного забезпечення.

Ключові слова	Java. Об'єктно-орієнтоване програмування (ООП). JVM (Java Virtual Machine). Гарбач-колектор (Garbage Collector). Колекції (Collections API). Потоки (Streams API). Багатопотоковість (Multithreading). JDBC. JPA. Hibernate. Spring Framework. REST API. Spring Boot. Тестування (JUnit, Mockito). CI/CD. Maven. Gradle. Серіалізація. Лямбда-вирази. Функціональне програмування. Інкапсуляція. Наслідування. Поліморфізм.
Формат курсу	Очний
	Проведення лекцій, практичних робіт та консультації для кращого розуміння тем
Теми	Див. СХЕМА КУРСУ
Підсумковий контроль, форма	Залік в кінці семестру в усній та практичній формі
Пререквізити	Для вивчення курсу студенти потребують базових знань з дисциплін «Алгоритмізація та програмування», «Об'єктно-орієнтоване програмування», «Операційні системи» та «Цифрова схемотехніка».
Навчальні методи та техніки, які будуть використовуватися під час викладання курсу	Лекції на основі презентацій, лабораторні роботи на основі методичок та їхній захист на основі індивідуальних звітів, виконання індивідуальних завдань та їхній захист.
Необхідне обладнання	Пристрої мультимедіа. ОС Windows 11, Visual Studio 2022, платформи MS Teams.
Критерії оцінювання (окремо для кожного виду навчальної діяльності)	<p>Оцінювання знань студентів здійснюється за 100-бальною шкалою. Максимальна кількість балів при оцінюванні знань впродовж семестру становить 100 балів.</p> <p>Програмою передбачено дві модульні контрольні роботи. Студент за одну контрольну роботу може отримати 25 балів (25x2 = 50).</p> <p>Програмою передбачено 9 практично-семінарських занять. За практичне заняття - від 2 до 5 балів . Кількість балів, які студент може отримати за усні відповіді подано в таблиці.</p> <p>За самостійну роботу впродовж семестру студент набирає 20 балів.</p>

Критерії оцінювання результатів навчання

90 – 100 б. – здобувач повно та ґрунтовно засвоїв всі теми навчальної програми, вміє вільно викладати зміст, має глибокі, міцні, систематичні знання всіх питань навчальної дисципліни, розуміє їх значення для своєї професійної підготовки; виконав усі завдання кожної теми та модульного поточного контролю.

81 – 89 б. - здобувач ґрамотно і по суті викладає програмний матеріал, застосовує теоретичні знання при виконанні практичних завдань, однак допускає незначні неточності, засвоїв більшість тем навчальної програми, вміє самостійно викладати зміст всіх питань навчальної дисципліни; виконав завдання кожної теми та модульного поточного контролю.

71 – 80 б. - здобувач добре знає програмний матеріал, володіє базовими навичками з виконання практичних завдань, самостійно обирає метод реалізації, але не завжди здатний провести аналіз і узагальнення результату, не завжди вміє вільно викладати зміст всіх питань навчальної дисципліни; виконав більшість завдань кожної теми та модульного поточного контролю.

61 – 70 б. – здобувач засвоїв тільки основний матеріал на рівні репродуктивного відтворення, але не знає окремих деталей, припускається неточностей, порушує послідовність у викладі матеріалу, може розв'язувати типові завдання за зразком, але допускає помилки, виконав окремі завдання кожної теми та модульного поточного контролю.

51 - 60 б. - здобувач відтворює основні поняття і визначення курсу, але досить поверхово, не виділяючи взаємозв'язок між ними, може сформулювати за допомогою викладача основні тези теми, допускає помилки, які повною мірою самостійно виправити не може; відчуває труднощі під час виконання практичних завдань, виконав лише деякі завдання кожної теми та модульного контролю.

21 – 50 б. – здобувач не знає значної частини програмного матеріалу, допускає суттєві помилки, з великими труднощами виконує практичні завдання, знайомий лише з деякими поняттями та визначеннями курсу; не виконав практичні завдання та завдання модульного поточного контролю.

	0 – 20 б. - необхідний повторний курс з навчальної дисципліни.
Питання до контрольних робіт	Білету для підсумкового заліку студенти отримують на останній лекції курсу.
Опитування	Анкету-оцінку з метою оцінювання якості курсу буде надано по завершенню курсу.

СХЕМА КУРСУ

Номер	Тема, план, короткі тези	Форма діяльності (заняття)	Література. Ресурси в Інтернеті	Завдання (лабораторна робота), год	Термін виконання
1	<p>Вступ до Java. Історія та особливості мови Java. Основи синтаксису.</p> <p>Тема знайомить студентів із мовою програмування Java, її історією, основними принципами та особливостями. Розглядаються ключові концепції Java, зокрема незалежність від платформи (Write Once, Run Anywhere), автоматичне керування пам'яттю, об'єктно-орієнтована модель. Вивчаються основи синтаксису Java: структура програми, змінні, типи даних, оператори, керуючі конструкції (умовні оператори, цикли), введення та виведення даних. Особлива увага приділяється принципам роботи JVM (Java Virtual Machine) та етапам виконання Java-програм.</p>	Лекція	1, 2, 3, 4, 5	Налаштування середовища та перша програма. Написання та запуск простої програми	<p>I - 14.02 21.02</p> <hr/> <p>II - 14.02 21.02</p>
2	<p>Змінні, типи даних і оператори. Примітивні та об'єктні типи. Операції та пріоритети. Робота з рядками (String, StringBuilder).</p> <p>Тема знайомить студентів із системою типізації в Java, особливостями змінних та їх використанням у програмах. Розглядаються примітивні типи даних (int, double, boolean, char) та об'єктні типи (String, масиви, колекції). Аналізується принципи роботи з операторами: арифметичні, логічні, порівняння, побітові та тернарний оператор, а також їх пріоритети під час виконання виразів.</p>	Лекція	1, 2, 3, 4, 5	Робота зі змінними та операторами. Використання різних типів даних	<p>I - 28.02 07.03</p> <hr/> <p>II - 28.02 07.03</p>
3	<p>Умовні конструкції та цикли. if-else, switch. Масиви та робота з ними</p>	Лекція	1, 2, 3, 4, 5	Реалізація умовних операторів та	

	<p>Тема знайомить студентів із механізмами керування потоком виконання програми в Java. Розглядаються умовні конструкції (if-else, switch), їхня логіка роботи та застосування для прийняття рішень у програмі. Аналізуються цикли (for, while, do-while), їхні особливості, варіанти використання та оптимізація повторюваних операцій. Окрема увага приділяється масивам як структурі даних, що дозволяє зберігати множину елементів одного типу. Розглядаються одно- та багатовимірні масиви, методи їхньої ініціалізації, сортування та основні операції над ними.</p>			циклів. Написання програм з використанням if-else, switch	<p>I - 14.03 18.03</p> <hr/> <p>II - 14.03 18.03</p>
4	<p>Функції та рекурсія. Методи, їх параметри та повернення значень. Тема знайомить студентів із принципами структурування коду за допомогою методів у Java. Розглядаються методи, їхнє визначення, виклик, передача параметрів та повернення значень. Аналізуються типи параметрів (значення та посилання), перевантаження методів та використання ключового слова return. Окрема увага приділяється рекурсії як способу реалізації алгоритмів, її перевагам і недолікам, а також порівнянню з ітеративними підходами. Розглядаються приклади ефективного використання методів для розбиття коду на логічні блоки та підвищення читабельності програм.</p>	Лекція	1, 2, 3, 4, 5	Функції та рекурсія. Створення методів. Модульна контрольна робота 1.	<p>I - 21.03 01.04</p> <hr/> <p>II - 21.03 01.04</p>
5	<p>Об'єктно-орієнтоване програмування в Java. Інкапсуляція, спадкування та поліморфізм. Тема знайомить студентів із основами об'єктно-орієнтованого програмування (ООП) в Java та його ключовими принципами. Розглядається інкапсуляція як механізм приховування деталей реалізації класів, використання модифікаторів доступу (private, protected, public) та методів getter і setter. Аналізується спадкування для повторного використання коду, ключове слово extends, базові та похідні класи. Окрема увага приділяється поліморфізму, перевизначенню (@Override), перевантаженню методів та динамічному зв'язуванню.</p>		1, 2, 3, 4	Об'єктно-орієнтоване програмування. Використання принципів ООП	<p>I - 04.04 15.03</p> <hr/> <p>II - 04.04 15.04</p>

	Розглядаються приклади застосування ООП у розробці гнучких та масштабованих програм.				
6	<p>Обробка винятків та введення/виведення. try-catch-finally блоки</p> <p>Тема знайомить студентів із механізмами обробки винятків у Java, що забезпечують стабільність та надійність програм.</p> <p>Розглядаються типи винятків (Checked, Unchecked, Errors), їхня класифікація та способи обробки.</p> <p>Аналізується використання блоків try-catch-finally для перехоплення помилок та забезпечення коректного завершення програмного процесу.</p> <p>Окрема увага приділяється механізмам генерації винятків (throw, throws), роботі з власними класами винятків.</p>	Лекція	1, 2, 3, 4	Обробка винятків та робота з файлами	<p>I - 18.04 29.04</p> <hr/> <p>II - 18.04 29.04</p>
7	<p>Колекції та робота з даними. Лямбда-вирази. Ітератори та стріми.</p> <p>Тема знайомить студентів із колекційною бібліотекою Java, що дозволяє ефективно працювати з наборами даних. Розглядаються основні інтерфейси та класи Java Collections Framework: List, Set, Map, їхні особливості, призначення та відмінності. Аналізується механізм ітераторів (Iterator, ListIterator) для обходу елементів колекцій. Окрема увага приділяється стрімам (Streams API) та методам функціонального програмування для обробки даних (map, filter, reduce). Розглядаються лямбда-вирази як спосіб скорочення коду та підвищення читабельності, їх застосування в Comparator, функціональних інтерфейсах та потоковій обробці даних.</p>	Лекція	1, 2, 3, 4	Колекції та лямбда-вирази. Робота зі списками та мапами	<p>I - 02.05 13.05</p> <hr/> <p>II - 02.05 13.05</p>
8	<p>Основи багатопотоковості та робота з мережею. Потоки та синхронізація. Робота з сокетами. ExecutorService.</p> <p>Тема знайомить студентів із багатопотоковістю в Java, принципами створення та управління потоками для підвищення продуктивності програм. Розглядається робота з класами Thread, Runnable, методи створення потоків, їх життєвий цикл та керування. Аналізуються механізми синхронізації потоків, використання synchronized, Lock,</p>	Лекція	1, 2, 3, 4	Багатопотоковість та мережеве програмування. Модульна контрольна робота 2.	<p>I - 16.05 27.05</p> <hr/> <p>II - 16.05 27.05</p>

	volatile для запобігання конфліктам при паралельному доступі до спільних ресурсів.				
9	Підведення підсумків курсу Обговорення екзаменаційних білетів.	Залік		Опрацювання білетів	I; II - 30.05