

ДЕРЖАВНИЙ ТОРГОВЕЛЬНО-ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ  
СИСТЕМА УПРАВЛІННЯ ЯКІСТЮ  
Система забезпечення якості освітньої діяльності та якості вищої освіти  
*сертифікована на відповідність ДСТУ ISO 9001:2015 / ISO 9001:2015*  
Кафедра інженерії програмного забезпечення та кібербезпеки

ПРОМИСЛОВЕ ПРОГРАМУВАННЯ JAVA/  
ENTERPRISE JAVA PROGRAMMING

СИЛАБУС/  
SILABUS

**ЗАТВЕРДЖЕНО**

засіданням кафедри



(протокол №. 1

від «07» серпня 2024 р.)

завідувач кафедри

 Олена КРИВОРУЧКО

Київ 2024

Назва освітньої компоненти	<b>ПРОМИСЛОВЕ ПРОГРАМУВАННЯ JAVA/ ENTERPRISE JAVA PROGRAMMING</b>
Спеціальність	121 «Інженерія програмного забезпечення»
Освітній ступінь	Другий (магістерський)
Освітньо-професійна програма	<b>ІНЖЕНЕРІЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ</b>
	<p>Лектор: <b>Цюцюра Микола</b></p> <p>-професор інженерії програмного забезпечення та кібербезпеки -доктор технічних наук -професор</p> <p>Резюме викладача: <a href="https://knute.edu.ua/blog/read/?pid=46090&amp;uk">https://knute.edu.ua/blog/read/?pid=46090&amp;uk</a> Науковий профіль: <a href="https://knute.edu.ua/blog/read/?pid=46717">https://knute.edu.ua/blog/read/?pid=46717</a> е-пошта: <a href="mailto:mitsiutsiura@knute.edu.ua">mitsiutsiura@knute.edu.ua</a></p>
	<p>Асистент лектора: <b>Сергій Бульба</b></p> <p>-ст.викладач, гарант освітньої програми «Інженерія програмного забезпечення» -к.е.н.</p> <p>Резюме викладача: <a href="https://knute.edu.ua/blog/read/?pid=48084&amp;uk">https://knute.edu.ua/blog/read/?pid=48084&amp;uk</a> Практична діяльність: Senior System Analyst, Solution Architect е-пошта: <a href="mailto:s.bulba@knute.edu.ua">s.bulba@knute.edu.ua</a></p>
Консультації	<a href="https://knute.edu.ua/blog/read/?pid=47103&amp;uk">https://knute.edu.ua/blog/read/?pid=47103&amp;uk</a>
Програма освітньої компоненти	<a href="https://knute.edu.ua/blog/read/?pid=48214">https://knute.edu.ua/blog/read/?pid=48214</a>
<b>ЗМІСТ ОСВІТНЬОЇ КОМПОНЕНТИ</b>	
Тема 1. Платформа Java – огляд та перспективи розвитку	Мета та основні принципи побудови платформи. Плюси та мінуси віртуальних машин. Типи платформ та їх редакціях. Основне призначення JavaEnterpriseEdition (EE). Архітектура JavaEE додатку. Вбудовані можливості та загально доступні бібліотеки. Подальший розвиток платформи Java
Тема 2. Вступ в	Віртуальна машина JVM, байт код, ClassLoader, системний завантажувач класів, відкладене

JVM, JRE, JDK та IDE	завантаження коду. Виконавче середовище Java (JavaRuntimeEnvironment – JRE). Комплект додатків розробника на мові Java (JavaDevelopmentKit – JDK). Набір програм и класів JDK. Структура Java програми. Особливості підготовки та виконання програм написаних на Java. Інтегровані середовища розробки на Java (IntegratedDevelopmentEnvironment – IDE).
Тема 3. Серіалізація в Java	Серіалізація-перетворення об'єкта у послідовність байтів. Стандартний механізм серіалізації. Призначення механізму серіалізації - передача даних з комп'ютера клієнта на сервер. Інтерфейс класу об'єкту java.io.Serializable. Серіалізація об'єкта методом writeObject класу java.io.ObjectOutputStream. Зворотній процес (десеріалізації) – метод readObject( ). Запис об'єкту у файл “temp.out” та структура цього файлу. Опис алгоритму серіалізації. Поняття маршалінгу та його відмість від терміну опис алгоритму серіалізації. Перетворення Java об'єкта в XML і навпаки. Процеси серіалізації та десеріалізації.
Тема 4. Концепція програмного інтерфейсу виклику віддалених методів (RemoteMethodInvocation - RMI) у мові Java	Структура пакету java.rmi. Алгоритм взаємодії між комп'ютерами при реалізації метода RMI. Аналіз об'єкту виключення який створюється при виникненні помилки у RMI. Створення та експортування по мережі серіалізованого об'єкту. Видалення об'єкту, якщо на нього ніхто не посилається. Наведення прикладу використання RMI.
Тема 5. Багатопоточне програмування на Java	Поняття багатопоточності та її необхідність. Відмінність між процесами та багатопоточністю. Процедура запуску та переривання завдання у окремому потоці. Поняття демон – потоку. Клас Thread– створення потоків та виконання операції з ними. Запуск потоку через метод publicvoidrun( ) інтерфейсу Runnable. Контроль стану потоків методами intgetState( ) та booleanisAlive класу Thread. Основні етапи роботи з потоком: створення, запуск, очікування завершення потоку та переривання потоку. Блокування (синхронізація) потоків. Монітори та умови потоку дозволяють передавати інформацію про його стан іншим зацікавленим потокам. Перетворення будь якого об'єкту в монітор. Передача управління іншому потоку.

	<p>Основні властивості моделі пам'яті: атомарність, видимість та упорядкованість.</p>
<p>Тема 6. Колекції інфраструктури CollectionsFramework</p>	<p>Загальне визначення колекції, як складна ієрархія інтерфейсів та класів, які представляють технологію управління групами об'єктів. Механізм роботи з колекціями. Методи, які розташовані на вершині колекції.</p> <p>Ієрархія успадкування списків та його класів. Методи класу ArrayList &lt;E&gt; та його методи.</p> <p>Клас двохспрямованого списку Deque&lt;E&gt; та його методи.</p> <p>Клас множений Set&lt;E&gt;, який використовує Хеш-таблицю для збереження колекції та його методи.</p> <p>Карта відображення об'єктів, зберігає пару «ключ-значень». Клас AbstractMap&lt;K,V&gt; та його методи.</p> <p>Клас Arrays, містить метод маніпулювання змістом масиву.</p> <p>Клас List, неупорядкована колекція, у якій допустимі значення дублюються.</p> <p>Створення узагальнення для класів та методів. Інтерфейси Comparable та Comparator. Сортування.</p>
<p>Тема 7. Лямбда функції в Java (функціональне програмування)</p>	<p>Поняття лямбди виразів. Питання які можна вирішувати за допомогою лямбда – виразів. Відмінність лямбда абстракцій від об'єктно-орієнтованого програмування. Особливості функціонального програмування. Перші лямбда вирази. Оператор стрілка.</p> <p>Різні способи запису лямбда – виразів. Ефективно фінальна змінна. Функціональні інтерфейси. Ромбовидний оператор «diamondoperator». Вбудовані функціональні інтерфейси. Інваріантні функції. Класи оболонки для примітивних типів даних. StreamAPI – прикладний інтерфейс для роботи з даними. Методи по замовченню та успадкування. Статичні методи у інтерфейсах.</p>
<p>Тема 8. JavaReflectionAPI.</p>	<p>Призначення рефлексії. Обмеження при роботі з рефлексії в Java. Забезпечення відображення в Java за допомогою класів Class та ClassLoader. Етапи освоєння техніки відображення: клас Class; застосування класів із пакету java.lang.reflect та створення нащадків ClassLoader.</p> <p>Створення екземпляру Class, який містить низькорівневу інформацію про цей клас. Основні операції з об'єктом Class. Знаходження суперкласів та інтерфейсів.</p>

	<p>Повернення масиву об'єктів типу Java та масиву всіх визначених полів об'єкта. Застосування методів set( ) та get( ) для зміни та отримання значень полів.</p> <p>Застосування методів forName( ) та newInstance( ) для динамічного завантаження і створення екземплярів класів у випадку, коли ім'я класу невизначеного до моменту виконання програми</p> <p>Повернення масиву типу методів за допомогою метода getMethodsoб'єкту Class.</p>
<p>Тема 9. Архітектура побудову розподілених додатків в Java на базі платформи JEE.</p>	<p>Загальна структура платформи JEE, яка побудована на базі Web-сервера Apache. Сервіс JNDI – універсальний сервіс збереження об'єктів у ієрархічній структурі імен.</p> <p>Ресурс DataSource–об'єкт, який дозволяє додатку отримати доступ до бази даних.</p> <p>Призначення сервлетів: читання явних даних, які передані з форм клієнта; читання неявних даних; генерація результатів; відправка клієнту явних даних у вигляді HTML; відправка неявних даних (статуси протоколу HTTP та заголовки).</p> <p>Налагодження та використання DataSource в ApacheTomcat. Приклади найпростіших сервлетів. Основні класи ServletAPI. Життєвий цикл сторінки та об'єкти сторінки.</p> <p>ПатернModelViewController (MVC). Використання патернаMVC в Web-приложенні.</p>
<p>Тема 10. Основи роботи у середовищі MySQLWorkbench</p>	<p>Послідовність встановлення пакету програм для Windows. Створення розширеної моделі сутності зв'язків (ExtendedEntityRelationshipModel –EERModel).</p> <p>Базові властивості полів: ім'я, тип даних, додаткові властивості полів. Створення полів та встановлення зв'язків з відображенням у EER – діаграмі.</p> <p>Створення фізичної моделі бази даних із EER - діаграми.</p> <p>Наповнення бази даних та модифікація даних здійснюється наступними засобами:</p> <p>1-й спосіб використовує команду INSERT, зміни даних UPDATE та видалення даних DELETE.</p> <p>2-й спосіб заповнення бази даних використовує візуальний засіб.</p> <p>3-й спосіб заповнення бази збереження даних у EER моделях.</p> <p>Використання команди SELECT для виконання запиту до бази даних: вибірка з однієї таблиці, використання умови відбору, використання функцій що агрегуються, упорядкування, підбиття підсумків по групам даних,</p>

	вибірка з декількох таблиць та створення таблиць за допомогою представлення.
Тема 11. Доступ до бази даних з Web-додатків	<p>Середовище розробки сервлетів: JDK, додаткова бібліотека <code>servlet-api.jar</code>, контейнер сервлетів, програма яка виконує функції Web-сервера. Встановлення JDK. Встановлення контейнера сервлетів Tomcat. Встановлення MySQLConnector/J для того щоб сервлети мали можливість взаємодіяти з базою даних.</p> <p>Етапи створення сервлетів: написання коду, компіляція сервлету, розміщення та реєстрація сервлету в контейнері. Запуск сервлету шляхом вводу його URL-адреси у адресному рядку Web-браузера.</p> <p>Підключення сервлету до бази даних за допомогою пакету <code>java.SDL.*</code>. Використання простих SQL-команд (побудованих з використанням метода <code>executeQuery()</code>), які не мають параметрів, що підставляються.</p> <p>Виконання динамічних SQL-команд, які містять підстановочні параметри побудованих за допомогою класу <code>PreparedStatement</code>.</p> <p>Застосування методів класу <code>SQLException</code> для отримання детальної інформації про помилки, які можуть виникати при використанні MySQL.</p>
<b>СПИСОК ОСНОВНИХ РЕКОМЕНДОВАНИХ ДЖЕРЕЛ</b>	
<ol style="list-style-type: none"> <li>1. Samoylov N. Learn Java 12 Programming: A step-by-step guide to learning essential concepts in Java SE 10, 11, and 12: Book / Nick Samoylov, – Packt Publishing. – 2019. – 690 p.</li> <li>2. Anbazhagan, Raulraja and Malawski, Konrad (Foreword by John Akerley). "Reactive Systems Architecture: Designing and Implementing an Entire Distributed System Using Akka." Manning Publications, 2021.</li> <li>3. Васильєв О.М. Програмування мовою Java / О.В. Васильєв. – Тернопіль: Навчальна книга – Богдан, 2022. – 696 с; іл.</li> <li>4. <b>Цензура М.О. Технологія JAVA. Навчальний посібник / М.О. Цензура, А.М. Десятко. – Київ: КНТЕУ, 2021. – 780 с.</b></li> </ol>	
<b>РЕЗУЛЬТАТИ ВИВЧЕННЯ ОСВІТНЬОЇ КОМПОНЕНТИ</b>	
Дисципліна забезпечує оволодіння здобувачами вищої освіти загальними та фаховими компетентностями і досягнення ними програмних результатів навчання:	
ЗК01.	Здатність до абстрактного мислення, аналізу та синтезу.
ЗК02.	Здатність спілкуватися іноземною мовою як усно, так і письмово.
СК 01.	Здатність аналізувати предметні області, формувати, класифікувати вимоги до програмного забезпечення.
СК 02.	Здатність розробляти і реалізовувати наукові та/або прикладні проекти у сфері інженерії програмного забезпечення.
СК 03.	Здатність проектувати архітектуру програмного забезпечення /

	програмного продукту, моделювати процеси функціонування окремих підсистем і модулів.
СК 05.	Здатність розробляти, аналізувати та застосовувати специфікації, стандарти, правила і рекомендації в сфері інженерії програмного забезпечення.
СК 09.	Здатність забезпечувати якість програмного забезпечення / програмного продукту.
РН01	Знати і застосовувати сучасні професійні стандарти і інші нормативно-правові документи з інженерії програмного забезпечення
РН02	Оцінювати і вибирати ефективні методи і моделі розроблення, впровадження, супроводу програмного забезпечення та управління відповідними процесами на всіх етапах життєвого циклу.
РН04	Виявляти інформаційні потреби і класифікувати дані для проєктування програмного забезпечення.
РН08	Розробляти і модифікувати архітектуру програмного забезпечення для реалізації вимог замовника.
РН09	Обґрунтовано вибирати парадигми і мови програмування для розроблення програмного забезпечення; застосовувати на практиці сучасні засоби розроблення програмного забезпечення.
РН10	Модифікувати існуючі та розробляти нові алгоритмічні рішення детального проєктування програмного забезпечення.
РН11	Забезпечувати якість на всіх стадіях життєвого циклу програмного забезпечення, у тому числі з використанням релевантних моделей та методів оцінювання, а також засобів автоматизованого тестування і верифікації програмного забезпечення.
РН13	Конфігурувати програмне забезпечення, керувати його змінами та розробленням програмної документації на всіх етапах життєвого циклу.
РН17	Збирати, аналізувати, оцінювати необхідну для розв'язання наукових і прикладних задач інформацію, використовуючи науково-технічну літературу, бази даних та інші джерела.

#### ОЦІНЮВАННЯ ЗНАНЬ ЗДОБУВАЧІВ ОСВІТИ

Сума балів, накопичених здобувачем вищої освіти за виконання всіх видів поточних навчальних завдань (робіт) на лабораторних/практичних заняттях, свідчить про ступінь оволодіння ним програмою освітньої компоненти на конкретному етапі її вивчення. Протягом семестру здобувачі освіти можуть набрати від 0 до 100 балів, що переводяться у національну шкалу оцінювання і відповідно у шкалу ЄКТС. Кількість балів відповідає певному рівню засвоєння дисципліни

#### Довідник з розподілу оцінок ДТЕУ (Шкала ЄКТС):

Бали ДТЕУ	Відсоток балів відносно загальної кількості одержаних прохідних балів	Кумулятивний відсоток отриманих прохідних балів
--------------	--	--

90-100	20	20
82-89	10	30
75-81	20	50
69-74	10	60
60-68	40	100

Розподіл балів за видами робіт:

Вид роботи	Бали	Вид роботи	Бали
Лабораторна робота 1	4	Самостійна робота 1	3
Лабораторна робота 2	4	Самостійна робота 2	3
Лабораторна робота 3	4	Самостійна робота 3	3
Лабораторна робота 4	4	Самостійна робота 4	3
Лабораторна робота 5	4	Самостійна робота 5	3
Лабораторна робота 6	4	Самостійна робота 6	3
Лабораторна робота 7	4	Самостійна робота 7	3
Лабораторна робота 8	4	Самостійна робота 8	3
Лабораторна робота 9	4	Самостійна робота 9	3
Лабораторна робота 10	4	Самостійна робота 10	3
Лабораторна робота 11	4	Самостійна робота 11	3
Захист проєкту	13	Наукова робота	10

Вимоги до критеріїв оцінювання самостійної роботи студента (оцінювання одного завдання у відсотковому еквіваленті)

40%	Детальний розгляд сутності та вмісту основних джерел. Подання фактів, ідей і результатів досліджень у логічній послідовності. Правильно проаналізовано поточний стан дослідження проблеми та зроблено огляд перспектив подальшого розвитку даного питання.
40%	Обґрунтованість аргументів, підтвердження особистого ставлення, пропозиції стосовно вирішення завдання, встановлення напрямків аналізу.
20%	Оформлення звіту у відповідності вимог

Критерії оцінювання самостійної роботи студента (оцінювання одного завдання у відсотковому еквіваленті)

100%	В повному обсязі володіє навчальним матеріалом, вільно самостійно та аргументовано його викладає під час усних виступів та письмових відповідей, глибоко та всебічно розкриває зміст теоретичних питань та лабораторних завдань, використовуючи при цьому обов'язкову та додаткову літературу. Правильно вирішив усі тестові завдання.
80%	Достатньо повно володіє навчальним матеріалом, обґрунтовано його викладає під час усних виступів та письмових відповідей, в основному розкриває зміст



	теоретичних питань та лабораторних завдань, використовуючи при цьому обов'язкову літературу. Але при викладанні деяких питань не вистачає достатньої глибини та аргументації, допускаються при цьому окремі несуттєві неточності та незначні помилки. Правильно вирішив більшість тестових завдань
60%	В цілому володіє навчальним матеріалом викладає його основний зміст під час усних виступів та письмових відповідей, але без глибокого всебічного аналізу, обґрунтування та аргументації, без використання необхідної літератури допускаючи при цьому окремі суттєві неточності та помилки. Правильно вирішив половину тестових завдань.
40%	Не в повному обсязі володіє навчальним матеріалом. Фрагментарно, поверхово (без аргументації та обґрунтування) викладає його під час усних виступів та письмових відповідей, недостатньо розкриває зміст теоретичних питань та лабораторних завдань, допускаючи при цьому суттєві неточності, правильно вирішив меншість тестових завдань.
20%	Частково володіє навчальним матеріалом не в змозі викласти зміст більшості питань теми під час усних виступів та письмових відповідей, допускаючи при цьому суттєві помилки. Правильно вирішив окремі тестові завдання.
0%	Не володіє навчальним матеріалом та не в змозі його викласти, не розуміє змісту теоретичних питань та практичних завдань. Не вирішив жодного тестового завдання.

#### ОСНОВНІ ПОЛОЖЕННЯ, ЩО РЕГЛАМЕНТУЮТЬ ОСВІТНІЙ ПРОЦЕС

діючі положення	<a href="https://knute.edu.ua/blog/read/?pid=44402">https://knute.edu.ua/blog/read/?pid=44402</a>
нормативно-правова база організації освітнього процесу	<a href="https://knute.edu.ua/blog/read/?pid=7330&amp;uk">https://knute.edu.ua/blog/read/?pid=7330&amp;uk</a>
студенту	<a href="https://knute.edu.ua/#forstudent">https://knute.edu.ua/#forstudent</a>

#### НЕФОРМАЛЬНА ОСВІТА

Рекомендовані сертифікаційні програми, курси, посібники користувача

Prometheus	<a href="https://prometheus.org.ua/course/course-v1:EPAM+JAVA101+2016_T2">https://prometheus.org.ua/course/course-v1:EPAM+JAVA101+2016_T2</a>
Codecademy	<a href="https://www.codecademy.com/learn/learn-java">https://www.codecademy.com/learn/learn-java</a>
MOOC	<a href="https://java-programming.mooc.fi">https://java-programming.mooc.fi</a>
30DayCoding	<a href="https://30dayscoding.com/blog/free-java-course-with-certificate-by-google">https://30dayscoding.com/blog/free-java-course-with-certificate-by-google</a>
Intellipaat	<a href="https://intellipaat.com/academy/course/java-">https://intellipaat.com/academy/course/java-</a>

	<a href="#">training/</a>
Sololearn	<a href="https://www.sololearn.com/en/learn/courses/java-introduction">https://www.sololearn.com/en/learn/courses/java-introduction</a>
<b>ПОЛІТИКА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ:</b>	
Відвідування лекційних та лабораторних занять: відвідування	Відвідування лекційних та лабораторних занять є обов'язковим. Допускаються пропуски занять з таких поважних причин, як хвороба (викладачу надається копія довідки від медичного закладу), участь в олімпіаді, творчому конкурсі тощо за попередньою домовленістю та згодою викладача за умови дозволу деканату (надаються документи чи інші матеріали, які підтверджують заявлену участь у діяльності студента).
Відпрацювання пропущених занять:	відпрацювання пропущених занять є обов'язковим незалежно від причини пропущеного заняття. Лекційне заняття має бути відпрацьоване до наступної лекції на консультації викладача з використанням ПЗ 365 Office Teams. Відпрацювання лекційного матеріалу передбачає вивчення пропущеного теоретичного матеріалу та складання тесту за цим матеріалом. Лабораторне заняття відпрацьовується під час консультації викладача (розклад консультацій на сайті).
Правила поведінки під час занять	обов'язковим є дотримання техніки безпеки в комп'ютерних лабораторіях. Студенти повинні приймати активну участь в обговоренні навчально матеріалу ознайомившись з ним напередодні (навчальний матеріал надається викладачем). Мобільні пристрої дозволяється використовувати лише під час онлайн тестування та підготовки практичних завдань в процесі заняття. Задля зручності, дозволяється використання ноутбуків та інших електронних пристроїв під час навчання в комп'ютерних аудиторіях (за взаємною згодою всіх учасників освітнього процесу)
Політика академічної доброчесності ДТЕУ	<a href="https://knute.edu.ua/blog/read/?pid=38987&amp;uk">https://knute.edu.ua/blog/read/?pid=38987&amp;uk</a>