

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ 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

ЗАТВЕРДЖЕНО

вченою радою
(пост. в. від 26.06.2021 р.)
Ректор



А. А. Мазаракі

**ENTERPRISE ПРОГРАМУВАННЯ JAVA /
ENTERPRISE JAVA PROGRAMMING**

**РОБОЧА ПРОГРАМА /
COURSE OUTLINE**

освітній ступінь	Магістр / Master
галузь знань	12 Інформаційні технології / Information Technologies
спеціальність	124 Системний аналіз / System Analysis
спеціалізація	Інформаційні технології та бізнес-аналітика (Data Science) / Information Technologies and Business Analytics (Data Science)

Київ 2021

Розповсюдження і тиражування без офіційного дозволу КНТЕУ заборонено

Автори: М.О. ЦЕНЗУРА, канд. техн. наук, доцент кафедри інженерії програмного забезпечення та кібербезпеки
Д. О. ТИЩЕНКО, канд. екон. наук, доцент кафедри інженерії програмного забезпечення та кібербезпеки
Н.О. КОТЕНКО, кандидат педагогічних наук, доцент кафедри інженерії програмного забезпечення та кібербезпеки
Д.Д. ГНАТЧЕНКО, асистент кафедри інженерії програмного забезпечення та кібербезпеки

Робочу програму розглянуто і затверджено на засіданні кафедри інженерії програмного забезпечення та кібербезпеки «25» травня 2021 р., протокол № 9.

Рецензенти: Савченко Т.В., к.т.н., доц. кафедри інженер програмного забезпечення та кібербезпеки
Бибешко Б.Т., Senior Software Engineer, Softorino Ltd

ENTERPRISE ПРОГРАМУВАННЯ JAVA / ENTERPRISE JAVA PROGRAMMING

РОБОЧА ПРОГРАМА / COURSE OUTLINE

освітній ступінь	магістр	/	Master
галузь знань	12 Інформаційні технології	/	Information Technologies
спеціальність	124 Системний аналіз	/	124 System Analysis
спеціалізація	Інформаційні технології та бізнес-аналітика (Data Science)	/	Information Technology and Business Analytics (Data Science)

1. СТРУКТУРА ДИСЦИПЛІНИ ТА РОЗПОДІЛ ГОДИН ЗА ТЕМАМИ (ТЕМАТИЧНИЙ ПЛАН)

Назва теми	Кількість годин				Форми контролю
	Всього годин/ кредитів	з них			
		Лекції	Лабораторні заняття / мод. контр.	Самостійна робота студентів	
Тема 1. Платформа Java – огляд та перспективи розвитку	16	2		14	УО
Тема 2. Вступ в JVM, JRE, JDK та IDE	16	2		14	УО
Тема 3. Сериалізація в Java	16	2		14	ІЗ
Тема 4. Концепція програмного інтерфейсу виклику віддалених методів (Remote Method Invocation - RMI) у мові Java	16	2		14	ІЗ
Тема 5. Багатопоточне програмування на Java	18	2	6	14	УО
Тема 6. Колекції інфраструктури Collections Framework	18	2	4	14	ІЗ
Тема 7. Лямбда функції в Java (функціональне програмування)	18	2		14	К
Тема 8. Java Reflection API	20	2	6	14	ІЗ
Тема 9. Архітектура побудову розподілених додатків в Java на базі платформи JEE	24	2	4	16	КР
Тема 10. Основи роботи у середовищі MySQL Workbench	18	2		12	ІЗ
Разом	180/6	20	20	140	
Підсумковий контроль	Екзамен письмовий				

Умовні позначення: УО – усне опитування; КР – контрольна робота; ІЗ – індивідуальне завдання.

2. ТЕМАТИКА ТА ЗМІСТ ЛЕКЦІЙНИХ ТА ЛАБОРАТОРНИХ ЗАНЯТЬ, САМОСТІЙНА РОБОТА СТУДЕНТІВ

Результати навчання	Навчальна діяльність	Робочий час студента (год.)
1	2	3
<p>Знати: призначення мови програмування Java; послідовність дій при створенні додатків за допомогою JDK; архітектуру Java EE додатку</p>	<p style="text-align: center;">Тема 1. Платформа Java – огляд та перспективи розвитку</p> <p style="text-align: center;">Тема 2. Вступ в JVM, JRE, JDK та IDE</p> <p>Лекція 1. Платформа Java – огляд та перспективи розвитку. Вступ в JVM, JRE, JDK та IDE</p> <p style="text-align: center;"><i>План лекції:</i></p> <ol style="list-style-type: none"> 1. Мета та основні принципи побудови платформи. 2. Архітектура Java EE додатку. 3. Подальший розвиток платформи Java. 4. Віртуальна машина JVM, байт код, Class Loader, системний завантажувач класів, відкладене завантаження коду. 5. Виконавче середовище Java (Java Runtime Environment – JRE). Комплект додатків розробника на мові Java (Development Kit – JDK). 6. Набір програм и класів JDK. Структура Java програми. Особливості підготовки та виконання програм написаних на Java. Інтегровані середовища розробки на Java (Integrated Development Environment – IDE). <p>Список рекомендованих джерел: <i>Основний: 1, 3, 5.</i> <i>Додатковий: 6, 8.</i> <i>Інтернет-ресурси: 15, 19.</i></p>	2
	<p>Завдання для самостійної роботи: Вивчення матеріалів лекції на основі самостійного опрацювання основних літературних джерел, зазначених у списку та електронних матеріалах, які надаються викладачем.</p> <p>Питання винесені на самостійне опрацювання:</p> <ol style="list-style-type: none"> 1. Плюси та мінуси віртуальних машин. 2. Типи платформ та їх редакціях. 3. Вбудовані можливості та загально доступні бібліотеки. 4. Основне призначення Java Enterprise Edition (EE). <p>Список рекомендованих джерел: <i>Основний: 1, 3, 5.</i> <i>Додатковий: 6, 8.</i> <i>Інтернет-ресурси: 15, 19.</i></p>	14

Результати навчання	Навчальна діяльність	Робочий час студента (год.)
1	2	3
<p>Знати: стандартний механізм серіалізації; призначення механізму; інтерфейс класу об'єкту java.io.Serializable; серіалізацію об'єкта методом writeObject класу java.io.ObjectOutputStream</p>	<p align="center">Тема 3. Серіалізація в Java</p> <p>Лекція 2. Серіалізація в Java</p> <p align="center"><i>План лекції:</i></p> <ol style="list-style-type: none"> 1. Серіалізація-перетворення об'єкта у послідовність байтів. 2. Стандартний механізм серіалізації. Призначення механізму серіалізації - передача даних з комп'ютера клієнта на сервер. 3. Інтерфейс класу об'єкту java.io.Serializable. 4. Серіалізація об'єкта методом writeObject класу java.io.ObjectOutputStream. 5. Зворотній процес (десеріалізації) – метод readObject(). 6. Запис об'єкту у файл «temp.out» та структура цього файлу. 7. Опис алгоритму серіалізації. <p>Список рекомендованих джерел: <i>Основний: 2, 4, 5. Додатковий: 7, 12. Інтернет-ресурси: 14, 20.</i></p> <p>Завдання для самостійної роботи: Вивчення матеріалів лекції на основі самостійного опрацювання основних літературних джерел, зазначених у списку та електронних матеріалів, які надаються викладачем.</p> <p>Питання винесені на самостійне опрацювання:</p> <ol style="list-style-type: none"> 1. Поняття маршалінгу та його відмість від терміну опис алгоритму серіалізації. 2. Перетворення Java об'єкта в XML і навпаки. 3. Процеси серіалізації та десеріалізації. <p>Список рекомендованих джерел: <i>Основний: 2, 4, 5. Додатковий: 7, 12. Інтернет-ресурси: 14, 20.</i></p>	<p align="center">2</p> <p align="center">14</p>
<p>Знати: структуру пакету java.rmi; алгоритм взаємодії між комп'ютерами при реалізації метода RMI</p>	<p align="center">Тема 4. Концепція програмного інтерфейсу виклику віддалених методів (Remote Method Invocation - RMI) у мові Java</p> <p>Лекція 3. Концепція програмного інтерфейсу виклику віддалених методів (Remote Method Invocation - RMI) у мові Java</p> <p align="center"><i>План лекції:</i></p> <ol style="list-style-type: none"> 1. Структура пакету java.rmi. 2. Алгоритм взаємодії між комп'ютерами при реалізації метода RMI. 3. Видалення об'єкту, якщо на нього ніхто не посилається. Наведення прикладу використання RMI. <p>Список рекомендованих джерел: <i>Основний: 1, 3, 4.</i></p>	<p align="center">2</p>

Результати навчання	Навчальна діяльність	Робочий час студента (год.)
1	2	3
	<p><i>Додатковий: 7, 9, 12.</i> <i>Інтернет-ресурси: 14, 20.</i></p> <p>Завдання для самостійної роботи: Вивчення матеріалів лекції на основі самостійного опрацювання основних літературних джерел, зазначених у списку та електронних матеріалів, які надаються викладачем.</p> <p>Питання винесені на самостійне опрацювання: 1. Аналіз об'єкту виключення який створюється при виникненні помилки у RMI. 2. Створення та експортування по мережі серіалізованого об'єкту.</p> <p>Список рекомендованих джерел: <i>Основний: 1, 3, 4.</i> <i>Додатковий: 7, 9, 12.</i> <i>Інтернет-ресурси: 14, 20.</i></p>	14
<p>Знати: поняття багатопоточності та її необхідність; відмінність між процесами та багатопоточністю; основні етапи роботи з потоком: створення, запуск, очікування завершення потоку та переривання потоку.</p> <p>Вміти: створювати потоки та виконувати операції з ними; здійснювати запуск потоку через метод <code>publicvoidrud()</code> інтерфейсу <code>Runnable</code>; контроль стану потоків методами <code>intgetState()</code> та</p>	<p>Тема 5. Багатопоточне програмування на Java Лекція 4. Багатопоточне програмування на Java <i>План лекції:</i></p> <ol style="list-style-type: none"> 1. Поняття багатопоточності та її необхідність. Відмінність між процесами та багатопоточністю. 2. Процедура запуску та переривання завдання у окремому потоці. Поняття демон – потоку. 3. Клас <code>Thrend</code> – створення потоків та виконання операції з ними. 4. Запуск потоку через метод <code>publicvoidrud()</code> інтерфейсу <code>Runnable</code>. 5. Контроль стану потоків методами <code>intgetState()</code> та <code>booleanisAlive</code> класу <code>Thrend</code>. 6. Основні етапи роботи з потоком: створення, запуск, очікування завершення потоку та переривання потоку. 7. Блокування (синхронізація) потоків. 8. Монітори та умови потоку дозволяють передавати інформацію про його стан іншим зацікавленим потокам. <p>Список рекомендованих джерел: <i>Основний: 3, 5.</i> <i>Додатковий: 8, 11, 12.</i> <i>Інтернет-ресурси: 14, 18.</i></p>	2
	<p>Завдання для самостійної роботи: Вивчення матеріалів лекції на основі самостійного опрацювання основних літературних джерел, зазначених у списку та електронних матеріалів, які надаються викладачем.</p> <p>Питання винесені на самостійне опрацювання: 1. Перетворення будь якого об'єкту в монітор. Передача управління іншому потоку.</p>	14

Результати навчання	Навчальна діяльність	Робочий час студента (год.)
1	2	3
booleanisAlive класу Thrend	<p>2. Основні властивості моделі пам'яті: атомарність, видимість та упорядкованість.</p> <p>Список рекомендованих джерел: <i>Основний: 3, 5.</i> <i>Додатковий: 8, 11, 12.</i> <i>Інтернет-ресурси: 14, 18.</i></p>	
	<p>Лабораторна робота 1. Реалізація багатопоточності в Java</p> <p><i>Мета:</i> навчитися реалізовувати багатопоточність використовуючи клас Thrend для створення потоків та виконання операцій з ними.</p> <p><i>Завдання:</i> у відповідності до індивідуального варіанту розробити додаток, який реалізовує багатопоточність.</p> <p><i>План заняття:</i></p> <ul style="list-style-type: none"> - Актуалізація теоретичного матеріалу. - Виконання завдань лабораторної роботи. - Презентація виконаної роботи. 	6
<p>Знати: 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>Вміти: застосовувати методи для роботи з колекціями</p>	<p>Тема 6. Колекції інфраструктури Collections Framework Лекція 5. Колекції інфраструктури Collections Framework</p> <p><i>План лекції:</i></p> <ol style="list-style-type: none"> 1. Загальне визначення колекції, як складна ієрархія інтерфейсів та класів, які представляють технологію управління групами об'єктів. 2. Механізм роботи з колекціями. Методи, які розташовані на вершині колекції. 3. Ієрархія успадкування списків та його класів. Методи класу ArrayList <E> та його методи. 4. Клас двохспрямованого списку Deque<E> та його методи. 5. Клас множений Set<E>, який використовує Хеш-таблицю для збереження колекції та його методи. 6. Карта відображення об'єктів, зберігає пару «ключ-значень». Клас AbstractMap<K,V> та його методи. 7. Клас Arrays, містить метод маніпулювання змістом масиву. 8. Клас List, неупорядкована колекція, у якій допустимі значення дублюються. <p>Список рекомендованих джерел: <i>Основний: 1, 4.</i> <i>Додатковий: 7, 10.</i> <i>Інтернет-ресурси: 14, 20.</i></p>	2
	<p>Завдання для самостійної роботи: Вивчення матеріалів лекції на основі самостійного опрацювання основних літературних джерел, зазначених у списку та електронних матеріалів, які надаються викладачем.</p> <p>Питання винесені на самостійне опрацювання:</p>	14

Результати навчання	Навчальна діяльність	Робочий час студента (год.)
1	2	3
	1. Створення узагальнення для класів та методів. 2. Інтерфейси Comparable та Comparator. Список рекомендованих джерел: <i>Основний: 1, 4.</i> <i>Додатковий: 7, 10.</i> <i>Інтернет-ресурси: 14, 20.</i>	
	Лабораторна робота 2. Робота з колекціями <i>Мета:</i> навчитися застосовувати методи для роботи з колекціями. <i>Завдання:</i> у відповідності до свого варіанту розширити роботу додатку реалізувавши в ньому колекції. Здійснити тестування ПП. <i>План заняття:</i> <ul style="list-style-type: none"> – Актуалізація теоретичного матеріалу. – Виконання завдань лабораторної роботи. – Презентація виконаної роботи. 	4
Знати: 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;	Тема 7. Лямбда функції в Java (функціональне програмування) Лекція 6. Лямбда функції в Java (функціональне програмування) <i>План лекції:</i> <ol style="list-style-type: none"> 1. Поняття лямбди виразів. Питання які можна вирішувати за допомогою лямбда – виразів. 2. Відмінність лямбда абстракцій від об’єктно-орієнтованого програмування. 3. Особливості функціонального програмування. Перші лямбда вирази. Оператор стрілка. 4. Різні способи запису лямбда – виразів. 5. Ефективно фінальна зміна. Функціональні інтерфейси. 6. Ромбовидний оператор «diamondoperator». 7. Вбудовані функціональні інтерфейси. Список рекомендованих джерел: <i>Основний: 3, 5.</i> <i>Додатковий: 11.</i> <i>Інтернет-ресурси: 14, 17.</i>	2

Результати навчання	Навчальна діяльність	Робочий час студента (год.)
1	2	3
<p>різні способи запису лямбда – виразів; функціональні інтерфейси</p>	<p>Завдання для самостійної роботи: Вивчення матеріалів лекції на основі самостійного опрацювання основних літературних джерел, зазначених у списку та електронних матеріалів, які надаються викладачем.</p> <p>Питання винесені на самостійне опрацювання:</p> <ol style="list-style-type: none"> 1. Інваріантні функції. 2. Класи оболонки для примітивних типів даних. 3. StreamAPI – прикладний інтерфейс для роботи з даними. 4. Методи по замовченню та успадкування. Статичні методи у інтерфейсах. <p>Список рекомендованих джерел: <i>Основний:</i> 3, 5. <i>Додатковий:</i> 11. <i>Інтернет-ресурси:</i> 14, 17.</p>	14
<p>Знати: 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; повернення масиву об'єктів типу Java та</p>	<p style="text-align: center;">Тема 8. Java Reflection API</p> <p>Лекція 7. Java Reflection API</p> <p style="text-align: center;"><i>План лекції:</i></p> <ol style="list-style-type: none"> 1. Призначення рефлексії. Обмеження при роботі з рефлексії в Java. 2. Забезпечення відображення в Java за допомогою класів Class та ClassLoader. 3. Етапи освоєння техніки відображення: клас Class; застосування класів із пакету java.lang.reflect та створення нащадків ClassLoader. 4. Повернення масиву об'єктів типу Java та масиву всіх визначених полів об'єкта. 5. Застосування методів set() та get() для зміни та отримання значень полів. 6. Застосування методів forName() та newInstance() для динамічного завантаження і створення екземплярів класів у випадку, коли ім'я класу невизначеного до моменту виконання програми. 7. Повернення масиву типу методів за допомогою метода getMethodsoб'єкту Class. <p>Список рекомендованих джерел: <i>Основний:</i> 1, 2. <i>Додатковий:</i> 9, 10. <i>Інтернет-ресурси:</i> 13, 18.</p>	2

Результати навчання	Навчальна діяльність	Робочий час студента (год.)
1	2	3
<p>масиву всіх визначених полів об'єкта; застосування методів set() та get() для зміни та отримання значень полів.</p> <p>Вміти: застосовувати Лямбда функції, працювати з класами Class та ClassLoader.</p>	<p>Завдання для самостійної роботи: Вивчення матеріалів лекції на основі самостійного опрацювання основних літературних джерел, зазначених у списку та електронних матеріалів, які надаються викладачем.</p> <p>Питання винесені на самостійне опрацювання: 1. Створення екземпляру Class, який містить низькорівневу інформацію про цей клас. 2. Основні операції з об'єктом Class. Знаходження суперкласів та інтерфейсів.</p> <p>Список рекомендованих джерел: <i>Основний:</i> 4. <i>Додатковий:</i> 5. <i>Інтернет-ресурси:</i> 11-14.</p> <p>Лабораторна робота 3. Лямбда функції та Java Reflection API <i>Мета:</i> застосовувати Лямбда функції, працювати з класами Class та ClassLoader. <i>Завдання:</i> у відповідності до свого варіанту розробити структуру класу, реалізувати клас Class та ClassLoader для забезпечення відображення та відповідні методи, застосувати лямбда функції.</p> <p><i>План заняття:</i> – Актуалізація теоретичного матеріалу. – Виконання завдань лабораторної роботи. – Презентація виконаної роботи.</p>	<p>14</p> <p>6</p>
<p>Знати: сервіс JNDI – універсальний сервіс збереження об'єктів у ієрархічній структурі імен; ресурс DataSource – об'єкт, який дозволяє додатку отримати доступ до бази даних; призначення сервлетів.</p> <p>Вміти: виконувати операції із розподіленими додатками в Java.</p>	<p>Тема 9. Архітектура побудови розподілених додатків в Java на базі платформи JEE</p> <p>Лекція 8. Архітектура побудови розподілених додатків в Java на базі платформи JEE</p> <p><i>План лекції:</i></p> <ol style="list-style-type: none"> 1. Загальна структура платформи JEE, яка побудована на базі Web-сервера Apache. 2. Сервіс JNDI – універсальний сервіс збереження об'єктів у ієрархічній структурі імен. 3. Ресурс DataSource – об'єкт, який дозволяє додатку отримати доступ до бази даних. 4. 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). 5. Патерн ModelViewController (MVC). Використання патерна MVC в Web-додатку. <p>Список рекомендованих джерел: <i>Основний:</i> 2, 5. <i>Додатковий:</i> 6, 9, 12.</p>	<p>2</p>

Результати навчання	Навчальна діяльність	Робочий час студента (год.)
1	2	3
	<p><i>Інтернет-ресурси: 18.</i></p> <p>Завдання для самостійної роботи: Вивчення лекції на основі самостійного опрацювання основних літературних джерел, зазначених у списку та електронних матеріалів, які надаються викладачем.</p> <p>Питання винесені на самостійне опрацювання:</p> <ol style="list-style-type: none"> 1. Налаштування та використання DataSource в ApacheTomcat. 2. Приклади найпростіших сервлетів. Основні класи ServletAPI. 3. Життєвий цикл сторінки та об'єкти сторінки. <p>Список рекомендованих джерел: <i>Основний: 2, 5. Додатковий: 6, 9, 12. Інтернет-ресурси: 18.</i></p> <p>Лабораторна робота 4. Робота із розподіленими додатками в Java <i>Мета:</i> навчитися створювати розподілені додатки. <i>Завдання:</i> у відповідності до свого варіанту розробити веб-додаток, який реалізовує роботу з базою даних. <i>План заняття:</i></p> <ul style="list-style-type: none"> - Актуалізація теоретичного матеріалу. - Виконання завдань лабораторної роботи. - Презентація виконаної роботи. 	14
<p>Знати: послідовність встановлення пакету програм для Windows; створення розширеної моделі сутності зв'язків (Extended Entity Relationship Model –EER Model); базові властивості полів: ім'я, тип даних, додаткові властивості полів; наповнення бази даних та</p>	<p>Тема 10. Основи роботи у середовищі MySQL Workbench</p> <p>Лекція 9. Основи роботи у середовищі MySQL Workbench <i>План лекції</i></p> <ol style="list-style-type: none"> 1. Послідовність встановлення пакету програм для Windows. Створення розширеної моделі сутності зв'язків (Extended Entity Relationship Model –EER Model). 2. Базові властивості полів: ім'я, тип даних, додаткові властивості полів. 3. Наповнення бази даних та модифікація даних 4. 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, вибірка з декількох таблиць та створення таблиць за допомогою представлення. <p>Список рекомендованих джерел: <i>Основний: 4, 5. Додатковий: 8, 11. Інтернет-ресурси: 13, 20.</i></p>	2

Результати навчання	Навчальна діяльність	Робочий час студента (год.)
1	2	3
модифікація даних	<p>Завдання для самостійної роботи: Вивчення матеріалів лекції на основі самостійного опрацювання основних літературних джерел, зазначених у списку та електронних матеріалів, які надаються викладачем.</p> <p>Питання винесені на самостійне опрацювання: 1. Створення полів та встановлення зв'язків з відображенням у EER – діаграмі. 2. Створення фізичної моделі бази даних із EER - діаграми.</p> <p>Список рекомендованих джерел: <i>Основний: 4, 5.</i> <i>Додатковий: 8, 11.</i> <i>Інтернет-ресурси: 13, 20.</i></p>	14
Разом		180
Підсумковий контроль - Екзамен письмовий		

Всі лекційні та практичні заняття виконуються на основі інтерактивних методів навчання у комп'ютерному середовищі.

СПИСОК РЕКОМЕНДОВАНИХ ДЖЕРЕЛ

Основний

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.
2. Sierra K. Head First Java: A Brain-Friendly Guide 3rd Edition: Book / Kathy Sierra, Bert Bates, Trisha Gee. - O'Reilly Media, 2021. - 688 p.
3. Urma R. Modern Java in Action: Lambdas, streams, functional and reactive programming 2nd Edition: Book / Raoul-Gabriel Urma, Mario Fusco, Alan Mycroft. - Manning; 2018. - 592 p.
4. Dascher S. Architecting Modern Java EE Applications. Designing lightweight, business – oriented enterprise applications in the age of cloud, containers, and Java EE 8. / S. Dascher. –Packt, Birmingham – Mumbai. 2017. – 384 p.
5. ¹Цензура М.О. *Технологія JAVA. Навчальний посібник* / М.О. Цензура, А.М. Десятко. – Київ: КНТЕУ, 2021. – 780 с.

Додатковий

6. Кадомський К.К. *Java. Теорія і практика* / Кадомський К.К., Ніколюк П.К. – Вінниця: Донну, 2019. – 197 с.
7. Ратушняк Т.В. *Програмування мовою JAVA: практикум* / Т. В. Ратушняк; Державна фіскальна служба України, Університет державної фіскальної служби України. – Ірпінь, 2017. – 212 с.
8. Спірінцева О.В. *В Java-технології та мобільні пристрої. Алгоритми і структури даних: навчальний посібник* / О.В. Спірінцева, О.А. Литвинов, В.В. Герасимов. – Д.: Вид-во ДНУ ім. О. Гончара, 2016. 140 с.
9. Ткаченко О. М. *Об'єктно-орієнтоване програмування мовою Java. Навчальний посібник.* / О. М. Ткаченко. – Вінниця: ВНТУ, 2016. – 107 с.
- 10.Троян С.О. *Програмування мовою Java: навч. посіб.* / С. О. Троян. – Умань: ФОП Жовтий О. О., 2017. – 132 с.
- 11.Цензура М.О. *Технологія JAVA: Методичні рекомендації для самостійної роботи студентів* / М. О. Цензура; Кафедра програмної інженерії та інформаційних систем; КНТЕУ; авт. Цензура М.О. – Київ : КНТЕУ, 2018 – 328 с.

Інтернет-ресурси

- 12.The Agile System Development Life Cycle (SDLC) – Режим доступу: <http://www.ambyssoft.com/essays/agileLifecycle.html> (дата звернення 24.12.19 р.).

13. Object Oriented Programming in Java. - Режим доступу: <https://www.coursera.org/specializations/object-oriented-programming> (дата завершення 25.10.21).
14. Introduction to Java . - Режим доступу: <https://www.coursera.org/learn/java-introduction> (дата завершення 25.10.21).
15. Java Підручник. - Режим доступу: <https://w3schoolsua.github.io/java/index.html> (дата завершення 27.10.21)
16. Основи програмування на Java. - Режим доступу: https://courses.prometheus.org.ua/courses/EPAM/JAVA101/2016_T2/about (дата завершення 27.10.21).
17. Java Is the Language of Possibilities. - Режим доступу: <https://www.oracle.com/java/technologies/> (дата завершення 27.10.21).
18. Олецкий О.В Перші кроки в Java . - Режим доступу: http://ekmair.ukma.edu.ua/bitstream/handle/123456789/13748/Oletskyi_Pershi_kroky_v_Java.pdf?sequence=1&isAllowed=y (дата завершення 27.10.21).

¹Курсивом виділені джерела, наявні в бібліотеці КНТЕУ