

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

ЗАТВЕРДЖУЮ

Декан факультету Інформаційних технологій

Тетяна Говорущенко

20 25 р.

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Технології програмування на Java

Призначення Робочої програми Для освітніх програм різних спеціальностей
Рівень вищої освіти Перший (бакалаврський)
Мова навчання Українська
Обсяг дисципліни, кредитів ЕКТС 8
Статус дисципліни Вибіркова
Факультет Інформаційних технологій
Кафедра Інженерії програмного забезпечення

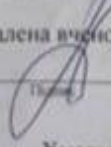
Форма здобуття освіти	Курс	Семестр	Обсяг дисципліни		Кількість годин						Форма семестрового контролю
			Кредити ЕКТС	Години	Аудиторні заняття					Самостійна робота (в т.ч. ПРС)	Залік
					Разом	Лекції	Лабораторні роботи	Практичні заняття	Семінарські заняття		
Д	2	4	8	240	82	32	50			158	+

Робоча програма складена на основі освітніх програм підготовки бакалавра та стандарту вищої освіти спеціальності.

Робоча програма складена  к.пед. н., доцент, **Наталія ПРАВОРСЬКА**
(Підпис) (прізвище) (Науковий ступінь, учасник, ім'я, прізвище, авторство)

Схвалена на засіданні кафедри **Інженерія програмного забезпечення**

Протокол від 28 серпня 2025 р. № 1 Зав. кафедри  **Леонід БЕДРАТЮК**
(Підпис) (Ім'я, прізвище)

Робоча програма розглянута та схвалена вченою радою факультету інформаційних технологій
Голова вченої ради факультету  **Тетяна ГОВОРУЩЕНКО**
(Підпис) (Ім'я, прізвище)

Хмельницький 2025

2 ЛИСТ ПОГОДЖЕННЯ

Посада	Назва факультету	Підпис	Ініціали, прізвище
Завідувач кафедри інженерії програмного забезпечення, доктор, фіз.-мат. наук, проф.	Факультет інформаційних технологій		Леонід БЕДРАТЮК
Гарант освітньо-професійної програми, доктор, фіз.-мат. наук, проф.	Факультет інформаційних технологій		Леонід БЕДРАТЮК

ТЕХНОЛОГІЇ ПРОГРАМУВАННЯ НА JAVA

Тип дисципліни	Вибіркова
Рівень вищої освіти	Перший
Мова викладання	Українська
Семестр	Четвертий
Кількість встановлених кредитів ЄКТС	8,0
Форми навчання, для яких викладається дисципліна	Денна

Результати навчання. Студент, який успішно завершив вивчення дисципліни, має: отримати певний рівень знання та набути практичні навички створення консольних та графічних застосувань (бібліотеки AVT та Swing) на мові Java; характеризувати моделі Java віртуальних машин, порівнювати функціональність компонентів JDK для подальшої роботи у середовищі; вміти перевіряти наявність середовища JRE у операційній системі; робити висновки про ефективність використання Java машин у порівнянні з іншими бібліотеками; збирати дані про контроль версій програмного забезпечення; роз'яснювати необхідність використання вбудованих класів; розрізняти інтерфейси та абстрактні класи; аналізувати код; передбачати появу додаткових викликів при використанні архітектури, що орієнтована на події; узагальнювати функціональність класів для використання компонентного підходу при розробці програм. Розробка людино-машинного інтерфейсу.

Процес вивчення дисципліни спрямований на **формування таких професійних компетенцій:**

- здатність створювати Java додатки на основі технічного завдання та підтримувати програмний продукт на протязі усього його життєвого циклу;
- здатність застосовувати завдання на основі вимог та побажань замовника з врахуванням технічного оснащення;
- вміти розробляти людино-машинний інтерфейс;
- здатність застосовувати об'єктно-орієнтовані підходи та засоби при створенні Java додатків;
- здатність здійснювати управління програмними потоками та використовувати різні технології програмування на мові Java;

Зміст навчальної дисципліни. Об'єктно-орієнтоване програмування у Java. Особливості створення додатків мовою JAVA. Середовище JRE. Компоненти JDK. Класи у Java. Наслідування. Підкласи. Суперкласи. Модікатор super. Інтерфейси. Визначення інтерфейсу. Пакети та технології Java Роль інтерфейсів у формуванні специфікацій. Інтерфейси для роботи з колекціями. Робота з текстовими файлами та стрічками (клас String) Java. Бібліотека Swing. Пакети та технології Java. Виняткові ситуації. Сокетне з'єднання по протоколу TCP / IP. Сервлети. Технологія підтримки динамічного web-контенту. Пакет javax.servlet. Життєвий цикл сервлета. Клас Toolkit з пакету java.awt. модель обробки зображень "постачальник-споживач" (Producer-Consumer). Робота з деревом каталогів. Підходи до повторного використання коду у Java.

Запланована навчальна діяльність. Мінімальний обсяг навчальних занять в одному кредиті ЄКТС навчальної дисципліни для *першого* (бакалаврського) рівня вищої освіти за денною формою здобуття освіти становить **8** годин на 1 кредит ЄКТС.

Форми (методи) навчання: лекції (з використанням методів проблемного навчання і візуалізації), лабораторні заняття (з використанням комп'ютерних методів та засобів навчання), самостійна робота.

Форми оцінювання результатів навчання: усне опитування, захист лабораторних робіт

Вид семестрового контролю: залік – 4 семестр.

Навчальні ресурси:

1. Cay Horstmann. Core Java: Fundamentals, Volume 1 (Oracle Press Java) 12th Edition. Oracle Press; 12th edition . 2021. – 944 p.
2. Yang Hu. Java Games Design Patterns: Learning Programming design patterns through games. On Kindle Scribe. 2020 – 204 p.
3. Herbert Schildt. Java: A Beginner's Guide, Ninth Edition 9th Edition. McGraw Hill; 9th 323 (January 7, 2022) - . 752 p.
4. Joshua Bloch. Java: Effective Programming (3rd Edition) Oracle Press - 2019- 1153 p.
5. Модульне середовище для навчання MOODLE. Доступ до ресурсу: <https://msn.khnu.km.ua>.

Викладач: кандидат педагогічних наук, доцент Праворська Н. І.

3. Пояснювальна записка

Дисципліна “Технології програмування на Java ” є дисципліною вільного вибору здобувача рівня бакалавр. Метою вивчення навчальних дисциплін є отримання студентами знань та набуття практичних навичок у галузі розробки, тестування та впровадження програмного забезпечення. Для вирішення теоретичних і практичних завдань, що виникають при діяльності людини у різних галузях науки, техніки та виробництва великий ефект дає використання обчислювальної техніки при умові достатнього програмного забезпечення. Відповідно знання і вміння користуватись сучасним програмним забезпеченням стає нагальною потребою для фахівця з вищою освітою. Тому дисципліна «Технології програмування на Java» у підготовці фахівців високої кваліфікації набуває особливо великого значення.

Пререквізити: введення в програмну інженерію, основи програмування, алгоритми і структури даних, об'єктно-орієнтоване програмування, іноземна мова (англійський).

Кореквізити: Web-програмування, прикладні інформаційні системи, основи командної розробки програмного забезпечення.

Мета дисципліни – надати студентам фундаментальні знання та практичні навички з базових концепцій мови програмування Java, що дозволяє розробляти надійні та ефективні застосунки. Ця дисципліна спрямована на вивчення принципів об'єктно-орієнтованого програмування, обробки винятків, роботи з колекціями та потоками, а також основ багатопоточності й управління пам'яттю. Дисципліна розвиває алгоритмічне мислення, допомагає розуміти внутрішні процеси JVM (Java Virtual Machine) і формує основи для вивчення більш складних технологій та фреймворків. Завдяки цьому студенти отримують необхідні знання для подальшого професійного розвитку в сфері Java-розробки та можуть писати масштабований, підтримуваний код.

Предмет дисципліни. Предмет дисципліни «Технології програмування на Java» охоплює теоретичні знання, задачі, функції та вимоги до програмних продуктів, принципи створення різних типів застосувань на Java.

Завдання дисципліни: ознайомлення студентів із особливостями програмування на мові Java та набуття їми досвіду у створенні Java застосунків.

Результати навчання. Використовувати базові знання інформатики й сучасних інформаційних систем та технологій, навички програмування, технології безпечної роботи в комп'ютерних мережах, методи створення баз даних та інтернет-ресурсів, технології розроблення алгоритмів і комп'ютерних програм мовами високого рівня із застосуванням об'єктно-орієнтованого програмування для розв'язання задач проектування і використання інформаційних систем та технологій. Проводити системний аналіз об'єктів проектування та обґрунтовувати вибір структури, алгоритмів та способів передачі інформації в інформаційних системах та технологіях. Аргументувати вибір програмних та технічних засобів для створення інформаційних систем та технологій на основі аналізу їх властивостей, призначення і технічних характеристик з урахуванням вимог до системи і експлуатаційних умов; мати навички налагодження та тестування програмних і технічних засобів інформаційних систем та технологій. Демонструвати знання сучасного рівня технологій інформаційних систем, практичні навички програмування та використання прикладних і спеціалізованих комп'ютерних систем та середовищ з метою їх запровадження у професійній діяльності. Обґрунтовувати вибір технічної структури та розробляти відповідне програмне забезпечення, що входить до складу інформаційних систем та технологій. Знати сучасні мови програмування та технології створення програмного забезпечення інформаційних систем та технологій

4. Структура залікових кредитів дисципліни “ Технології програмування на Java ”

Теми	Лекції	Лаб. роб.	Самостійна робота студентів
Тема 1. Предмет і зміст дисципліни. Синтаксис та структура мови Java.	2	6	12
Тема 2. Стандартні бібліотеки класів в мові Java.	2		12
Тема 3. Виняткові ситуації. Введення/виведення в Java	2	6	12
Тема 4. Робота з інтерфейсами. Серіалізація об'єктів.	2		12
Тема 5. Події. Бібліотека Swing.	2	6	12
Тема 6. Ідентифікація типу під час виконання. Множинні нитки виконання (Multiple threads).	2		14
Тема.7. Технологія JavaBeans. Взаємодія з SQL-сервером	2	6	14
Тема 8. Введення в мережеві протоколи та мережева взаємодія. Обмін даними через мережу Internet	2		14
Тема 9. Робота з аудіо та відео	4	6	14
Тема 10. Поняття шаблонів проєктування. Породжуючі шаблони проєктування	4	6	14
Тема 11. Поняття шаблонів проєктування. Структурні шаблони проєктування	4	6	14
Тема 12. Поняття шаблонів проєктування. Поведінкові патерни проєктування	4	8	14
Разом за 4-й семестр	32	50	158

5. ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ “ Технології програмування на Java ”

5.1. Зміст лекційного курсу*

№ лекції	Перелік тем лекцій, їх анотація	Кількість годин
Тема 1. Предмет і зміст дисципліни. . Синтаксис та структура мови Java.		
1	Лекція 1. Основні поняття і принципи ООП в мові Java. Вступ. Java – об'єктно-орієнтована мова програмування. Принципи ООП. Посилки на об'єкти. Поля класу і змінні програми. Область видимості і час життя змінних. Опис методів класу та виклик. Доступ до полів класу. Передача параметрів. Реалізація принципів об'єктно-орієнтованого підходу в Java. Описувачі обмеження доступу. Повернення значення в методі класу. Ключове слово this. Статичні поля і методи класу. Знайомство з документацією. Основні конструкції мови Java. Операції. Літерали. Оператори. Масиви. Конструктори класів. Робота з рядками Література: [1] стор. 24-49; [2] стор. 58-112; [3] стор. 45-136; [5] стор. 32-116; [6] стор. 31-168; [7] стор. 32-74; [8] стор. 34-89; [9] стор. 28-94; [10] стор. 41-112; [12] стор. 27-113.	2
Тема 2. Стандартні бібліотеки класів в мові Java.		
2	Лекція 2. Робота з бібліотеками та пакетами. Робота з класами (вкладеними, статичними). Колекції об'єктів, множини та робота з масивами. Знайомство з бібліотеками та пакетами. Перша діалогова	2

	програма. Спадкування класів. Вкладені класи. Статичні вкладені класи. Колекції об'єктів. Завдання порядку на множині. Колекції - асоціативні масиви (Map). Сортювання і пошук. Література: [2] стор. 113-149; [3] стор. 136-158; [5] стор. 119-134; [8] стор. 91-149; [10] стор. 115-191.	
Тема 3. Виняткові ситуації. Введення/виведення в Java		
3	Лекція 3. Виняткових ситуацій та робота потоками введення/виведення в мові Java. Обробка виняткових ситуацій (Exceptions). Виняткові ситуації (продовження). Введення/виведення (I/O). Ієрархія Writer. Література: [2] стор. 149-178; [4] стор. 129-210; [8] стор. 150-167.	2
Тема 4. Робота з інтерфейсами. Серіалізація об'єктів.		
4	Лекція 4. Робота з інтерфейсами. Серіалізація в мові Java. Робота з класами доступу до файлів. Інтерфейс FileFilter. Клас JFileChooser. Поняття серіалізації. Серіалізація об'єктів. Інтерфейс Serializable. Класи ObjectOutputStream і ObjectInputStream. Література: [1] стор. 301-376; [4] стор. 213-298; [9] стор. 302-386.	2
Тема 5. Події. Бібліотека Swing.		
5	Лекція 5. Робота з подіями та бібліотекою Swing в мові Java. Події та їх реалізація. Подієва модель в JFC. Слухачі подій. Події та потоки. Клас JList бібліотеки Swing. Зовнішній вигляд відображуваного списку. Література: [5] стор. 321-376; [7] стор. 212-246; [9] стор. 387-425.	2
Тема 6. Ідентифікація типу під час виконання. Множинні нитки виконання (Multiple threads).		
6	Лекція 6. Ідентифікація типу під час виконання в мові Java. Рефлексія. Ідентифікація типу під час виконання. Базові засоби RTTI. Визначення типу об'єкта в програмі. Створення об'єктів за допомогою методу newInstance. Рефлексія. Механізм рефлексії. Література: [2] стор. 179-212; [3] стор. 161-189; [7] стор. 247-256; [9] 425-469. Множинні нитки виконання (Multiple threads). Процес організації нитки. Обережність при використанні ниток. Література: [3] стор. 191-213; [6] стор. 151-179; [7] стор. 247-269. Самореєстрація ниток. Завершення і зупинка нитки. Інтерфейс Runnable. Нитки - демони (Daemons). Пріоритети ниток. Диспетчеризація ниток. Конкурентний доступ до ресурсів при багатопотокової обробці. Засоби синхронізації ниток в Java. Література: [3] стор. 213-234; [7] стор. 270-283, [9] стор. 469-491.	2
Тема.7. Технологія JavaBeans. Взаємодія з SQL-сервером		
7	Лекція 7. Основні поняття технології JavaBeans. Основні поняття. Що таке Bean. Властивості бінов (Bean properties). Інтроекція бінов за допомогою reflection API. Пов'язані властивості (bound properties) і події. Література: [5] стор. 377-395; [8] стор. 213-289; [10] стор. 212-259. Робота зі зв'язаними властивостями. Створення і використання зв'язаної властивості. Обмежені властивості (constrained properties). Література: [7] стор. 283-320; [8] стор. 289-316; [10] стор. 260-301.	2
Тема 8. Введення в мережеві протоколи та мережева взаємодія. Обмін даними через мережу Internet		
8	Лекція 8. Основні поняття мережевих протоколів в мові Java. Основи моделі OSI. Підмережі, маска мережі. Порти. Робота з адресами. Передача даних по протоколу TCP. Передача даних по протоколу UDP. Література: [2] стор. 391-415; [10] стор. 302-359; [12] стор. 316-363. Передача даних при мережевій взаємодії в мові Java. Робота з URL-з'єднаннями. Обмін даними через мережу Internet. Функція connect(). Клас Socket. Клас InetAddress. Література: [3] стор. 236-316; [5] стор. 398-421; [9] стор. 501-526.	2

Тема 9. Робота з аудіо та відео		
9	Лекція 9. Поняття мультимедіа. Огляд найпоширеніших аудіо- та відео-форматів. Медіаконтейнери. Програвачі-плеєри. Утиліти-конвертери мультимедіа. Грабери. Конвертація форматів. Бібліотеки Java для роботи з аудіо- та відео форматами. Література: [19]; [23]; [25]	2
10	Лекція 10. Робота з основними класами для створення аудіо-та відео файлів у Java. Класи Manager та Player. Клас ControllerListener та робота з ними при створенні Java-програм. Література: [19]; [23]; [25]	2
Тема 10. Поняття шаблонів проектування. Породжуючі шаблони проектування.		
11	Лекція 11. Шаблони проектування в мові Java. Вступ в шаблони проектування. Класифікація шаблонів проектування (шаблони GoF). Шаблони Abstract Factory. Builder, Factory Method,. Література: [9] стор. 532-549; [10] стор. 360-410.	2
12	Лекція 12. Шаблони проектування в мові Java. Шаблони проектування Prototype, Singleton Література: [10] стор. 411-463; [12] стор. 364-379, [19], [23].	2
Тема 11. Поняття шаблонів проектування. Структурні шаблони проектування		
13	Лекція 13. Структурні шаблони проектування. Шаблон Адаптер (Wrapper, Обгортка, Adapter). Шаблон Міст (Bridge). Шаблон Компонувальник (Дерево, Composite). Література: [10] стор. 465-486; [11] стор. 163-178; [19], [23].	2
14	Лекція 14. Структурні шаблони проектування. Шаблон Декоратор (Wrapper, Обгортка, Decorator). Шаблон Фасад (Facade). Шаблон Легковаговик (Пристосуванець, Кэш, Flyweight). Література: [10] стор. 487-516; [11] стор. 178-203; [19], [23].	2
Тема 12. Поняття шаблонів проектування. Поведінкові патерни проектування		
15	Лекція 15. Поняття шаблонів проектування. Поведінкові патерни проектування. Ланцюжок обов'язків (CoR, Chain of Command, Chain of Responsibility). Шаблон Команда (Действие, Транзакция, Action, Command). Шаблон Ітератор (Iterator). Шаблон Посередник (Intermediary, Controller, Mediator). Шаблон Знімок (Зберігач, Memento). Література: [10] стор. 519-587; [12] стор. 381-451 [19], [23]	2
16	Лекція 16. Поняття шаблонів проектування. Поведінкові патерни проектування. Шаблон Спостерігач (Видавець-підписник, Слухач, Observer). Шаблон Стан (State). Шаблон Стратегія (Strategy). Шаблон Шаблонний метод (Pattern Method). Шаблон Відвідувач (Visitor). Література: [10] стор. 588-723; [12] стор. 452-531; [19], [23]	2
Разом за 4-й семестр		32

5.2 Зміст лабораторних занять

№ з/п	Теми лабораторних занять	Години
1	Базові операції та оператори Java. Масиви та файли	6
2	Візуальні компоненти.	6
3	Зв'язні структури в Java	6
4	Багатопотокові додатки	6
5	Взаємодія з SQL-сервером	6
6	Обмін даними через мережу Internet	6
7	Робота з аудіо та відео	6
8	Робота з патернами Abstract Factory, Builder	8

Разом за 4-й семестр	50
----------------------	----

5.3 Зміст самостійної роботи

Об'єм самостійної роботи з дисципліни «Технології програмування на Java» становить 158 години. Вони включають опрацювання лекційного матеріалу, теоретичних і лабораторних завдань, підготовку до виконання лабораторних робіт, їх захисту і поточне тестування.

Самостійна робота здобувачів вищої освіти полягає у систематичному опрацюванні програмного матеріалу з відповідних джерел інформації, підготовці до лабораторних занять, виконанні індивідуальних завдань, тестування з теоретичного матеріалу тощо. Крім цього до послуг студентів сторінка навчальної дисципліни у Модульному середовищі для навчання, де розміщені Робоча програма дисципліни та необхідні документи з її навчально-методичного забезпечення.

Номер тижня	Назва теми	Години
1	Тема 1. Предмет і зміст дисципліни. Синтаксис та структура мови Java. Опрацювання лекційного матеріалу, виконання лабораторної роботи №1. Література: [1] стор. 24-49; [2] стор. 58-112; [3] стор. 45-136; [5] стор. 32-116; [6] стор. 31-168; [7] стор. 32-74; [8] стор. 34-89; [9] стор. 28-94; [10] стор. 41-112; [12] стор. 27-113.	8
2	Тема 2. Стандартні бібліотеки класів в мові Java. Опрацювання лекційного матеріалу. Захист лабораторної роботи №1. Література: [2] стор. 113-149; [3] стор. 136-158; [5] стор. 119-134; [8] стор. 91-149; [10] стор. 115-191.	10
3	Тема 3. Виняткові ситуації. Введення/виведення в Java. Опрацювання лекційного матеріалу, виконання лабораторної роботи №2. Література: [2] стор. 149-178; [4] стор. 129-210; [8] стор. 150-167.	10
4	Тема 4. Робота з інтерфейсами. Серіалізація об'єктів. Опрацювання лекційного матеріалу. Захист лабораторної роботи №2. Література: [1] стор. 301-376; [4] стор. 213-298; [9] стор. 302-386.	10
5	Тема 5. Події. Бібліотека Swing. Опрацювання лекційного матеріалу. Виконання лабораторної роботи №3. Підготовка до проходження тесту. Література: [5] стор. 321-376; [7] стор. 212-246; [9] стор. 387-425.	10
6	Тема 6. Ідентифікація типу під час виконання. Множинні нитки виконання (Multiple threads). Опрацювання лекційного матеріалу. Захист лабораторної роботи № 3. Література: [2] стор. 179-212; [3] стор. 161-189; [7] стор. 247-256; [9] 425-469; [3] стор. 191-213; [6] стор. 151-179; [7] стор. 247-269; [3] стор. 213-234; [7] стор. 270-283, [9] стор. 469-491. Тестування	10
7	Тема 7. Технологія JavaBeans. Взаємодія з SQL-сервером. Опрацювання лекційного матеріалу. Виконання лабораторної роботи №4. Література: [5] стор. 377-395; [8] стор. 213-289; [10] стор. 212-259; Література: [7] стор. 283-320; [8] стор. 289-316; [10] стор. 260-301.	10
8	Тема 8. Введення в мережеві протоколи та мережева взаємодія. Обмін даними через мережу Internet. Опрацювання лекційного матеріалу. Захист лабораторної роботи №4. Література: [2] стор. 391-415; [10] стор. 302-359; [12] стор. 316-363.	10
9	Тема 8. Введення в мережеві протоколи та мережева взаємодія. Обмін даними через мережу Internet. Опрацювання лекційного матеріалу. Виконання лабораторної роботи №5. Література: [3] стор. 236-316; [5] стор. 398-421; [9] стор. 501-526.	10
10	Тема 9. Робота з аудіо та відео. Опрацювання лекційного матеріалу. Захист лабораторної роботи №5. Література: [19]; [23]; [25]	10

11	Тема 10. Поняття шаблонів проектування. Породжуючі шаблони проектування. Опрацювання лекційного матеріалу, виконання лабораторної роботи №6. Література: [9] стор. 532-549; [10] стор. 360-410.	10
12	Тема 10. Поняття шаблонів проектування. Породжуючі шаблони проектування. Опрацювання лекційного матеріалу, Захист лабораторної роботи №6. Література: [10] стор. 411-463; [12] стор. 364-379, [19], [23].	10
13	Тема 11. Поняття шаблонів проектування. Структурні шаблони проектування. Опрацювання лекційного матеріалу. Виконання лабораторної роботи №7. Література: [10] стор. 465-486; [11] стор. 163-178; [19], [23].	10
14	Тема 11. Поняття шаблонів проектування. Структурні шаблони проектування. Опрацювання лекційного матеріалу. Захист лабораторної роботи №7. Література: [10] стор. 487-516; [11] стор. 178-203; [19], [23].	10
15	Тема 12. Поняття шаблонів проектування. Поведінкові патерни проектування. Опрацювання лекційного матеріалу. Виконання лабораторної роботи №8. Література: [10] стор. 519-587; [12] стор. 381-451 [19], [23]	10
16	Тема 12. Поняття шаблонів проектування. Поведінкові патерни проектування. Опрацювання лекційного матеріалу. Захист лабораторної роботи №8. Література: [10] стор. 588-723; [12] стор. 452-531; [19], [23]	10
Разом за 4-й семестр		158

На самостійне опрацювання студентів виносяться визначені у методичних рекомендаціях до лабораторних занять та самостійної роботи питання з кожної теми. Керівництво самостійною роботою та контроль за виконанням індивідуального завдання здійснюється викладачем згідно з розкладом консультацій у позаурочний час.

Вимоги до виконання контрольної роботи (для студентів заочної форми здобуття освіти) та індивідуального домашнього завдання (для студентів денної форми здобуття освіти) викладені в Модульному середовищі для навчання на сторінці навчальної дисципліни.

6. Технології та методи навчання

Процес навчання з дисципліни ґрунтується на використанні традиційних та сучасних технологій, зокрема: лекції (з використанням методів проблемного навчання і візуалізації); практичні заняття (бесіда, демонстрування, спостереження, з використанням кейсів, розв'язування задач, презентацій), самостійна робота (індивідуальне домашнє завдання, опрацювання теоретичного матеріалу).

Процес навчання з дисципліни ґрунтується на використанні традиційних та сучасних технологій та методів навчання, зокрема: методи навчання за джерелом передачі і сприймання інформації (словесні (пояснення, дискусія, консультування), практичні (інструктування, розв'язування ситуаційних задач), наочні (демонстрування, ілюстрування, спостереження); за логікою передачі і сприймання навчальної інформації; за рівнем самостійності пізнавальної діяльності (методи проблемного викладу, частково пошукові, дослідницькі); методи стимулювання і мотивації учіння, інтерактивні; метод аналізу конкретних ситуацій (case-study) з використанням технологій візуалізації, інформаційно-комунікаційних та технології дистанційного навчання (сервіс для проведення онлайн конференцій Zoom, Модульне середовище для навчання тощо).

7. Методи контролю

Поточний контроль здійснюється під час практичних занять, а також у дні проведення контрольних заходів, встановлених робочою програмою і графіком освітнього процесу.

При цьому використовуються такі методи поточного контролю:

- усне опитування;
- тестовий контроль теоретичного матеріалу;
- оцінювання результатів роботи на лабораторних заняттях (розв'язування задач, участь у обговоренні ситуацій, самостійні роботи);

– оцінювання результатів виконання індивідуального домашнього завдання.

При виведенні підсумкової семестрової оцінки враховуються результати як поточного контролю, так і підсумкового контролю, який проводиться з усього матеріалу дисципліни за білетами, попередньо розробленими і затвердженими на засіданні кафедри. Здобувач вищої освіти, який набрав з будь-якого виду навчальної роботи, суму балів нижчу за 60 відсотків від максимального балу, не допускається до семестрового контролю, поки не виконає обсяг роботи, передбачений Робочою програмою. Здобувач вищої освіти, який набрав позитивний середньозважений бал (60 відсотків і більше від максимального балу) з усіх видів поточного контролю і не склав іспит, вважається таким, який має академічну заборгованість. Ліквідація академічної заборгованості із семестрового контролю здійснюється у період екзаменаційної сесії або за графіком, встановленим деканатом відповідно до «Положення про контроль і оцінювання результатів навчання здобувачів вищої освіти у ХНУ».

8. Політика дисципліни

Політика навчальної дисципліни загалом визначається системою вимог до здобувача вищої освіти, що передбачені чинними положеннями Університету про організацію і навчально-методичне забезпечення освітнього процесу. Зокрема, проходження інструктажу з техніки безпеки; відвідування занять з дисципліни є обов'язковим. За об'єктивних причин (підтверджених документально) теоретичне навчання за погодженням із лектором може відбуватись в індивідуальному режимі. Успішне опанування дисципліни і формування фахових компетентностей і програмних результатів навчання передбачає необхідність підготовки до практичних занять (вивчення теоретичного матеріалу з теми, активно працювати на занятті, розв'язувати задачі, брати участь у дискусіях щодо прийнятих рішень при виконанні здобувачами задач).

Здобувачі вищої освіти зобов'язані дотримуватися термінів виконання усіх видів робіт у встановлені терміни, передбачені робочою програмою навчальної дисципліни. Пропущене практичне заняття здобувач зобов'язаний відпрацювати у встановлений викладачем термін, але не пізніше, ніж за два тижні до кінця теоретичних занять у семестрі.

Засвоєння студентом теоретичного матеріалу з дисципліни оцінюється за результатами опитування під час практичних занять, тестування й виконання індивідуального домашнього завдання. Виконання індивідуального завдання завершується його здачею на перевірку у терміни, встановлені графіком самостійної роботи. У якості ІДЗ здобувач може підготувати реферат або тези доповіді на конференцію за однією з тем навчальної дисципліни при дотриманні узгоджених з викладачем термінів його виконання.

Здобувач вищої освіти, виконуючи самостійну роботу з дисципліни, має дотримуватися політики доброчесності (заборонені списування, підказки, плагіат, використання штучного інтелекту (без вірного цитування)). У разі порушення політики академічної доброчесності в будь-яких видах навчальної роботи здобувач вищої освіти отримує незадовільну оцінку і має повторно виконати завдання з відповідної теми (виду роботи), що передбачені робочою програмою. Будь-які форми порушення академічної доброчесності під час вивчення навчальної дисципліни не допускаються та не толеруються.

У межах вивчення навчальної дисципліни здобувачам вищої освіти передбачено визнання і зарахування результатів навчання, набутих шляхом неформальної освіти, що розміщені на доступних платформах (<https://github.com>, <https://docs.oracle.com/javase/tutorial/java/index.html>, <http://www.bruceeckel.com/>, <http://java.sun.com/products/jdk/1.3/docs/index.html>, <http://java.sun.com/products/jdk/1.3/devdocs-vs-specs.html>, <http://developer.java.sun.com/developer/onlineTraining/>, <http://java.sun.com/products/java-media/jmf/2.1.1/guide/JMFTOC.html>), які сприяють формуванню компетентностей і поглибленню результатів навчання, визначених робочою програмою дисципліни, або забезпечують вивчення відповідної теми та/або виду робіт з програми навчальної дисципліни (детальніше у Положенні про порядок визнання та зарахування результатів навчання здобувачів вищої освіти у ХНУ).

9. Оцінювання результатів навчання студентів у семестрі

Оцінювання академічних досягнень здобувача вищої освіти здійснюється відповідно до «Положення про контроль і оцінювання результатів навчання здобувачів вищої освіти у ХНУ».

При поточному оцінюванні виконаної здобувачем роботи з кожної структурної одиниці і отриманих ним результатів викладач виставляє йому певну кількість балів із призначених робочою програмою для цього виду роботи. При цьому кожна структурна одиниця (робота) може бути зарахована, якщо здобувач набрав не менше 60 відсотків (мінімальний рівень для позитивної оцінки) від максимально можливої суми балів, призначеної структурній одиниці.

Будь-які форми порушення академічної доброчесності **не допускаються та не толеруються**.

Отриманий здобувачем бал за зарахований вид навчальної роботи (структурну одиницю) після її оцінювання викладач виставляє в електронному журналі обліку успішності здобувачів вищої освіти. За умови виконання усіх видів навчальної роботи за результатами поточного контролю протягом вивчення навчальної дисципліни, встановлених її Робочою програмою, здобувач денної форми здобуття освіти з навчальної дисципліни, підсумковим контролем для якої є іспит, може набрати до 60 балів (здобувач заочної форми – до 50 балів). Позитивну підсумкову оцінку здобувач може отримати, якщо за результатами поточного та підсумкового контролів набере від 60 до 100 балів. Семестрова підсумкова оцінка розраховується в автоматизованому режимі в інформаційній підсистемі «Електронний журнал» (ІС «Електронний університет») і відповідно до накопиченої суми балів визначається оцінка за інституційною шкалою та шкалою ЄКТС (див. таблицю Співвідношення...), яка заноситься в екзаменаційну відомість, а також до Індивідуального навчального плану здобувача вищої освіти.

Таблиця – Критерії оцінювання навчальних досягнень здобувача вищої освіти

Оцінка та рівень досягнення здобувачем запланованих ПРН та сформованих компетентностей	Узагальнений зміст критерія оцінювання
Відмінно (високий)	Здобувач вищої освіти глибоко і у повному обсязі опанував зміст навчального матеріалу, легко в ньому орієнтується і вміло використовує понятійний апарат; уміє пов'язувати теорію з практикою, вирішувати практичні завдання, впевнено висловлювати і обґрунтовувати свої судження. Відмінна оцінка передбачає логічний виклад відповіді мовою викладання (в усній або у письмовій формі), демонструє якісне оформлення роботи і володіння спеціальними приладами та інструментами, прикладними програмами. Здобувач не вагається при видозміні запитання, вміє робити детальні та узагальнюючі висновки, демонструє практичні навички з вирішення фахових завдань. При відповіді допустив дві–три несуттєві <i>помилки</i> .
Добре (середній)	Здобувач вищої освіти виявив повне засвоєння навчального матеріалу, володіє понятійним апаратом, орієнтується у вивченому матеріалі; свідомо використовує теоретичні знання для вирішення практичних задач; виклад відповіді грамотний, але у змісті і формі відповіді можуть мати місце окремі неточності, нечіткі формулювання правил, закономірностей тощо. Відповідь здобувача вищої освіти будується на основі самостійного мислення. Здобувач вищої освіти у відповіді допустив дві–три <i>несуттєві помилки</i> .
Задовільно (достатній)	Здобувач вищої освіти виявив знання основного програмного матеріалу в обсязі, необхідному для подальшого навчання та практичної діяльності за професією, справляється з виконанням практичних завдань, передбачених програмою. Як правило, відповідь здобувача вищої освіти будується на рівні репродуктивного мислення, здобувач вищої освіти має слабкі знання структури навчальної дисципліни, допускає неточності і <i>суттєві помилки</i> у відповіді, вагається при відповіді на видозмінене запитання. Разом з тим, набув навичок, необхідних для виконання нескладних практичних завдань, які відповідають мінімальним критеріям оцінювання і володіє знаннями, що дозволяють йому під керівництвом викладача усунути неточності у відповіді.
Незадовільно (недостатній)	Здобувач вищої освіти виявив розрізнені, безсистемні знання, не вміє виділяти головне і другорядне, допускається помилок у визначенні понять, перекручує їх зміст, хаотично і невпевнено викладає матеріал, не може використовувати знання при вирішенні практичних завдань. Як правило, оцінка «незадовільно» виставляється здобувачеві вищої освіти, який не може продовжити навчання без додаткової роботи з

	вивчення навчальної дисципліни.
--	---------------------------------

Структурування дисципліни за видами навчальної роботи і оцінювання результатів навчання студентів денної форми здобуття освіти у семестрі

Аудиторна робота								Контрольні заходи	Самостійна робота	Семестровий контроль	Разом
<i>Червний семестр</i>											
Лабораторні роботи №:								Тестовий контроль:	Індивідуальне звання	Залік	Сума балів
1	2	3	4	5	6	7	8	Т			
Кількість балів за вид навчальної роботи (мінімум-максимум)											
3-5	3-5	3-5	3-5	3-5	3-5	3-5	3-5	3-5	3-5		
24-40								24-30	12-30		60-100

Умовні позначення: Т – тема навчальної дисципліни

Оцінювання результатів захисту лабораторної роботи

Виконана й оформлена відповідно до встановлених Методичними рекомендаціями вимог лабораторна робота комплексно оцінюється викладачем при її захисті з урахуванням таких критеріїв: самостійність та правильність виконання; повнота відповіді та знання методики проектування та розробки коду (застосунку); наявність відповідей на контрольні питання до відповідної лабораторної роботи, наявність звіту зі скріпками коду та результатів його роботи, тощо.

Результат виконання і захисту здобувачем вищої освіти кожної лабораторної роботи оцінюється відповідно до таблиці Критеріїв оцінювання навчальних досягнень здобувача вищої освіти та рівня досягнення здобувачем запланованих ПРН та сформованих компетентностей з присвоєнням йому відповідної суми балів.

У випадку виявлення здобувачем рівня знань, нижчого ніж 60 відсотків від максимального балу, встановленого Робочою програмою для кожної структурної одиниці, лабораторна робота йому не зараховується і для її захисту він має детальніше опрацювати матеріал з теми роботи, методику її виконання, виправити грубі помилки та повторно вийти на її захист у призначений для цього викладачем час.

Оцінювання результатів навчання з теоретичного матеріалу (тестовий контроль)

Тест, передбачений Робочою програмою, складається із 30 тестових завдань, кожне з яких є рівнозначним. Максимальна сума балів, яку може набрати студент, складає 100 (*кількість набраних балів за тестове завдання може бути різною*).

Відповідно до таблиці структурування видів робіт за тематичний контроль здобувач залежно від кількості правильних відповідей може отримати від 60 до 100 балів.

Розподіл балів в залежності від наданих правильних відповідей на тестові завдання

Кількість правильних відповідей	1-17	18-23	24-26	27-30
Відсоток правильних відповідей	0-59	60-79	80-89	90-100
Отримана оцінка	-	3	4	5
Кількість балів	>24	24	27	30

На тестування відводиться 30 хвилин. Студент проходить тестування в он-лайн режимі у Модульному середовищі для навчання. При отриманні негативної оцінки тест слід перездати до терміну *наступного* контролю.

Оцінювання результатів виконання індивідуального завдання

Виконане та оформлене відповідно до вимог, визначених Методичними рекомендаціями, індивідуальне домашнє завдання (ІДЗ) комплексно оцінюється викладачем з урахуванням таких

критеріїв: самостійність виконання; правильність розв'язання поставлених задач; обґрунтованість вибору методів розв'язання; повнота пояснень та аргументованість відповідей; якість оформлення та дотримання вимог до структури і змісту роботи.

Результат виконання здобувачем вищої освіти кожного ІДЗ оцінюється відповідно до таблиці **Критеріїв оцінювання навчальних досягнень здобувача вищої освіти** з урахуванням рівня досягнення запланованих програмних результатів навчання та сформованих компетентностей. За підсумками захисту присвоюється відповідна сума балів (мінімальний позитивний бал – 3 бали, максимальний – 5 балів).

У разі, якщо здобувач вищої освіти виявив рівень знань і виконання ІДЗ, що нижчий ніж 60 відсотків від максимальної кількості балів, встановленої Робочою програмою для цієї структурної одиниці, завдання не зараховується. У такому випадку студент має повторно опрацювати зміст завдання, усунути помилки та здати на перевірку доопрацьоване ІДЗ у терміни, погоджені з викладачем.

Розподіл балів в залежності від отриманої оцінки за індивідуальне завдання

Отримана оцінка	> 3	3	4	5
Відсотки відповідно до отриманої оцінки	> 60	60	80	100
Отриманий бал	>12	12	21	30

Підсумкова семестрова оцінка за інституційною шкалою і шкалою ЄКТС визначається в автоматизованому режимі після внесення викладачем результатів оцінювання у балах з усіх видів навчальної роботи до електронного журналу. Співвідношення інституційної шкали оцінювання і шкали оцінювання ЄКТС наведені нижче у таблиці «Співвідношення».

Семестровий залік виставляється на останньому занятті за умови якщо загальна сума балів, яку накопичив здобувач з дисципліни (іншого освітнього компонента) за результатами *поточного* контролю, знаходиться у межах від 60 до 100 балів. При цьому за інституційною шкалою ставиться оцінка «*зараховано*», а за шкалою ЄКТС – буквене позначення оцінки, що відповідає набраній студентом сумі балів відповідно до таблиці Співвідношення. Присутність здобувача у цьому випадку не є обов'язковою.

Таблиця – Співвідношення інституційної шкали оцінювання і шкали оцінювання ЄКТС

Оцінка ЄКТС	Рейтингова шкала балів	Інституційна оцінка (рівень досягнення здобувачем вищої освіти запланованих результатів навчання з навчальної дисципліни)	
		Залік	Іспит/диференційований залік
A	90-100	Зараховано	Відмінно/Excellent – високий рівень досягнення запланованих результатів навчання з навчальної дисципліни, що свідчить про безумовну готовність здобувача до подальшого навчання та/або професійної діяльності за фахом
B	83-89		Добре/Good – середній (максимально достатній) рівень досягнення запланованих результатів навчання з навчальної дисципліни та готовності до подальшого навчання та/або професійної діяльності за фахом
C	73-82		Задовільно/Satisfactory – Наявні мінімально достатні для подальшого навчання та/або професійної діяльності за фахом результати навчання з навчальної дисципліни
D	66-72		
E	60-65		
FX	40-59	Незараховано	Незадовільно/Fail – Низка запланованих результатів навчання з навчальної дисципліни відсутня. Рівень набутих результатів навчання є недостатнім для подальшого навчання та/або професійної діяльності за фахом
F	0-39		Незадовільно/Fail – Результати навчання відсутні

10. Питання для самоперевірки

“Технології програмування на Java”

1. Особливості мови і платформи Java.
2. Класифікація програм за типом виконання (компільовані, що інтерпретуються, виконувані на віртуальних машинах). Віртуальна машина Java. JIT-компіляція.
3. Створення найпростішої програми на Java, її компіляція в байт-код і запуск.
4. Засоби розробки Java-додатків. Інтегровані середовища розробки.
5. Вбудовані типи даних. Способи завдання літералів різних типів.
6. Приведення типів (явне і автоматичне). Константи і змінні.
7. Оператор присвоювання. Порядок дій (пріоритет операторів).
8. Арифметичні оператори. Оператори інкремента і декремента.
9. Вбудований клас Math. Псевдовипадкові числа.
10. Оператори порівняння і логічні оператори.
11. Оператори розгалуження. Умовний оператор. Мінімізація кількості перевірок.
12. Оператори розгалуження. Оператор множинного вибору. Його порівняння з умовним оператором.
13. Вбудований клас String. Строкові операції.
14. Стандартні потоки вводу-виводу. Організація введення і виведення даних. Клас Scanner.
15. Оператори організації циклів. Цикл типу «n раз».
16. Оператори організації циклів. Цикл типу «поки» (з перед- і постпроверкою умови).
17. Масиви. Способи оголошення і ініціалізації масивів. Індексція і розмір масиву.
18. Масиви. Алгоритми сортування. Багатовимірні масиви.
19. Статичні методи класів. Методи функціонального і процедурного типу.
20. Сигнатура методу. Перевантаження методів.
21. Процедурне програмування. Об'єктно-орієнтоване програмування. Порівняння парадигм.
22. Основні поняття ООП. Об'єкти і класи. Абстракція даних. Сценарій побудови об'єктно-орієнтованої програми.
23. Члени класів. Методи і поля.
24. Спеціальні методи класів (конструктори). Конструктор за замовчуванням.
25. Модифікатори рівня доступу (default, public, protected, private).
26. Основні принципи ООП. Інкапсуляція.
27. Основні принципи ООП. Спадкування. Управління спадкуванням.
28. Інтерфейси як засіб реалізації множинного спадкоємства.
29. Основні принципи ООП. Поліморфізм. Засоби реалізації поліморфізму.
30. Ієрархія класів Java. Корінний клас Object і його методи.
31. Виняткові ситуації. Обробка виняткових ситуацій.
32. Додатки з графічним інтерфейсом з використанням GUI-пакетів і аплетів.

11. Питання до тесту

“Технології програмування на Java”

1. Назвати основні блоки в Java-програмах.
2. Що містять в собі пакети.
3. Дати визначення первинному класу.
4. Перелічити стандартні типи даних в Java.
5. Описати клас Date.
6. За допомогою яких функцій відбувається виведення даних в Java-програмах.
7. Дати визначення обов'язковому параметру перетворення/
8. За допомогою яких функцій відбувається введення даних в Java-програмах?
9. Дати ви значення масиву.
10. Що виконує оператор new?
11. Назвати переваги масивів в мові Java.
12. Як визначаються багатовимірні масиви в мові Java?
13. Назвати категорії виразів, які закінчуються "крапка с комою".
14. Назвати оператори гілкування в мові Java.
15. Назвати оператори циклів в мові Java.
16. Для чого використовуються мітки і оператор break в мові Java?

17. Як створити клас в мові Java?
18. Як проводити тестування в програмі, написаній на мові Java?
19. Дати визначення власному типу у мові програмування Java.
20. Дати визначення конструктору.
21. Як оголосити клас з атрибутами в мові Java?
22. Дати визначення рядку в мові Java.
23. Які класи підтримують роботу з рядками в мові Java?
24. Назвати особливості класу String, методи класу String для роботи з рядками.
25. Для чого використовуються класи StringBuilder та StringBuffer?
26. Як встановити розмір буферу?
27. Назвати методи роботи з рядками в класах StringBuilder та StringBuffer?
28. Дати визначення стандартній бібліотеці в програмуванні.
29. З чого складається стандартна бібліотека.
30. Дати визначення пакету. Які пакети містить стандартна бібліотека Java.
31. Що треба зробити, щоб використати клас з пакету?
32. Описати основні класи пакета java.lang.
33. Що собою представляють класи System і String?
34. Дати визначення колекції.
35. Що містить колекція Collection<E>, Set<E>, Queue<E>?
36. Назвати методи для роботи з колекціями.
37. Дати визначення виключенню.
38. Назвати недоліки AWT.
39. Назвати основні стандартні LookAndFeel компоненти.
40. Що містить SWT?
41. Дати визначення компоненту і контейнеру.
42. Назвати засоби для створення користувацьких інтерфейсів в Java.
43. Описати пакет AWT.
44. Описати пакет Swing.
45. Дати визначення аплету.
46. Описати призначення бібліотеки Swing.
47. Описати модель подій в Swing.
48. Дати визначення колекції.
49. Назвати і описати основні класи колекцій.
50. Назвати найпоширеніший варіант колекції.
51. Назвати методи колекції в класі List.
52. Що виконує методи в класі Collections.
53. Що забезпечують об'єкти типу File?
54. Що забезпечують потоки введення-виведення?
55. Описати призначення класу RandomAccessFil.
56. Назвати способи для створення класу, екземплярами якого будуть потоки виконання.
57. Описати метод створення класу успадкуванням.
58. Описати метод створення класу реалізуванням інтерфейсу.
59. Назвати найважливіші константи та методи класу Thread.
60. Назвати найважливіші методи об'єктів типу Thread.
61. Назвати додаткові методи, які будуть корисними при написанні багатопотокової програми.
62. Що забезпечують програми, які написані з використанням багатопоточності?
63. Дати визначення шаблону та описати його призначення.
64. Описати класифікацію шаблонів.
65. Описати породжуючі шаблони проектування.
66. Описати шаблон Abstract Factory
67. Описати шаблон Builder
68. Описати шаблон Factory Method
69. Описати шаблон Prototype
70. Описати шаблон Singleton
71. Описати структурні шаблони.
72. Описати поведінкові шаблони.

73. Описати шаблони для конкуруючих операцій.
74. Описати проєктування на базі об'єктів.
75. Описати шаблони домена.
76. Описати шаблони джерела даних.
77. Дати визначення нитки (потoku) в програмуванні.
78. Дати визначення мультипотokuвoї обробці.
79. Описати найпростіший спосіб створення нитки.
80. Який клас використовується для створення багатопоточності?
81. Як реалізувати інтерфейс Runnable.
82. Як відбувається завершення й зупинка нитки.
83. Пояснити розподіл пріоритетів для ниток.
84. Пояснити, як є засоби синхронізації ниток в Java.
85. Пояснити, як застосовується багатопоточність для анімації

12. Навчально-методичне забезпечення

Освітній процес з дисципліни «Технології програмування на Java» повністю і в достатній кількості забезпечений необхідною навчально-методичною літературою. Зокрема, викладачами кафебри підготовлені і видані такі роботи:

– Електронний аналог друкованого видання Технології програмування на Java: методичні вказівки до виконання лабораторних робіт для студентів спеціальності 121 «Інженерія програмного забезпечення». Сертифікат №3є/20. Затверджено рішенням науково-методичної ради університету протокол №8 від 18.06.2020 р.

13. Матеріально-технічне та програмне забезпечення дисципліни (за потреби)

Інформаційна та комп'ютерна підтримка: ПК, планшет, смартфон або інший мобільний пристрій, проєктор. Програмне забезпечення: програми Microsoft Office або аналогічні, доступ до мережі Інтернет, робота з презентаціями.

Вивчення навчальної дисципліни не потребує використання спеціального програмного прикладного забезпечення, крім загальноновживаних програм і операційних систем.

14. РЕКОМЕНДОВАНА ЛІТЕРАТУРА

ОСНОВНА:

1. Joshua Bloch. Java: Effective Programming (3rd Edition) Oracle Press – 2019 – pp. 1153
2. Brian Goetz, Tim Peierls, Joshua Bloch, Joseph Bowbeer, David Holmes, Doug Lea. Java Concurrency in Practice 1st Edicion. Addison-Wesley Professional – 2020 – pp. 432
3. Chris Mayfield, Think Java: How To Think Like a Computer Scientist - O'Reilly Media; 2nd edition (January 7, 2020) – pp. 323
4. David Flanagan, Benjamin J. Evans. Java in a Nutshell: A Desktop Quick Reference 7th Edition - O'Reilly Media – 2019 – pp. 458
5. Cay Horstmann. Core Java: Fundamentals, Volume 1 (Oracle Press Java) 12th Edition. Oracle Press; 12th edition . 2021. – pp. 944
6. Tony Gaddis, Godfrey Muganda. Starting Out with Java: From Control Structures through Data Structures (What's New in Computer Science). Pearson; 4th edition – pp. 1408
7. Yang Hu. Java Games Design Patterns: Learning Programming design patterns through games. On Kindle Scribe. 2020 – pp.204
8. Herbert Schildt. Java: A Beginner's Guide, Ninth Edition 9th Edition. McGraw Hill; 9th 323 (January 7, 2022) – pp..752

ДОПОМІЖНА:

9. Cay Horstmann. Core Java Volume I--Fundamentals (Core Series) 11th Edition. Pearson; –2018 – pp.. 928
10. Weiss, Mark Allen. Data structures and algorithm analysis in Java / Mark Allen Weiss. – 3rd ed.- pp. 636

11. Herbert Schildt. Java: A Beginner's Guide, Eighth Edition 8th. McGraw Hill; 8th edition (November 2, 2018) – pp.. 720
12. Eric Freeman, Elisabeth Robson, Bert Bates, Kathy Sierra. Head First Design Patterns: A Brain-Friendly Guide. – 2018 – pp. 656
13. Ken Kousen. Modern Java Recipes: Simple Solutions to Difficult Problems in Java 8 and 9 1st Edition. O'Reilly Media – 2017 – pp. 322
14. Robert Lafore. Data Structures and Algorithms in Java 2nd Edition. Sams Publishing – 2010 – pp.. 800
15. David Flanagan. Java Examples in a Nutshell, 3rd Edition. O'Reilly Media – 2004 – pp. 701
16. Cay S. Horstmann, Gary Cornell. Core Jav Volume I – Fundamentals. Oracle. – 2013 – pp.. 1588
17. Cay S. Horstmann, Gary Cornell. Core Jav Volume II – Fundamentals. Oracle. – 2015 – pp.. 1232
18. Тьюторіал по Java <https://docs.oracle.com/javase/tutorial/>
19. The Java Tutorials: Trail: Learning the Java Language <https://docs.oracle.com/javase/tutorial/java/index.html>
20. "Thinking in Java", Bruce Eckel. <http://www.bruceeckel.com/>.
21. Документація від Sun <http://java.sun.com/products/jdk/1.3/docs/index.html>.
22. Корткий опис JAVA <http://java.sun.com/products/jdk/1.3/devdocs-vs-specs.html>.
23. Навчальні курси для розробників <http://developer.java.sun.com/developer/onlineTraining/>.
24. <http://java.sun.com/products/java-media/jmf/2.1.1/guide/JMFTOC.html>
25. Праворська Н. І. Зручне програмне забезпечення для бізнесу з можливістю адаптації до ринкових вимог. Вісник Хмельницького національного університету, Том 1, №1, 2023 (317) – С. 257-262

15. ІНФОРМАЦІЙНІ РЕСУРСИ

1. Модульне середовище для навчання. Доступ до ресурсу: <https://msn.khmn.u.edu.ua/course/view.php?id=7040>
2. Електронна бібліотека університету . Доступ до ресурсу: <http://library.khmn.u.edu.ua/>
3. Репозитарій ХНУ. Доступ до ресурсу: <http://elar.khmn.u.edu.ua>
4. GitHub. Навчальні проекти Java. [Електронний ресурс]. – Режим доступу <https://github.com/CamelNotFemale/Java>
5. GitHub. Запитання для співбесіди на Java Developer. – Режим доступу : <https://github.com/enhorse/java-interview>