


МІНІСТЕРСТВО ФІНАНСІВ УКРАЇНИ
ІРПІНСЬКИЙ ФАХОВИЙ КОЛЕДЖ ЕКОНОМІКИ ТА ПРАВА

ЗАТВЕРДЖУЮ

Заступник директора
з навчальної роботи


Олена ПУСТОВА

« 01 » 03 2023 р.

РОБОЧА ПРОГРАМА

навчальної дисципліни

«Об'єктно-орієнтоване програмування»

для підготовки фахового молодшого бакалавра

за галуззю знань 12 Інформаційні технології

спеціальності 122 «Комп'ютерні науки»

Робоча програма навчальної дисципліни «Об'єктно-орієнтоване програмування» складено на основі навчальної програми, затвердженої у 2023 році.

Розробник

Тетяна РАТУШНЯК

Розглянуто і схвалено на засіданні циклової комісії інформаційних технологій, математичних та природничих дисциплін, протокол № 02 від 31 серпня 2023 року.

Голова циклової комісії



Анастасія ШЦУК

Завідувач навчально-методичного кабінету коледжу



Людмила РОМАНЕНКО

ЗМІСТ

ПЕРЕДМОВА	4
КРИТЕРІЇ ОЦІНЮВАННЯ.....	5
ЗМІСТ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ	7
СПИСОК РЕКОМЕНДОВАНИХ ДЖЕРЕЛ ІНФОРМАЦІЇ.....	16

ПЕРЕДМОВА

В сучасних умовах неможливо уявити фахівця будь-якої галузі без уміння використання комп'ютерної техніки. Особливо це стосується спеціалістів, чия діяльність безпосередньо пов'язана з використанням ПК. Створення сучасних програмних продуктів вимагає досконалого знання мов програмування як високого так низького рівня, а також уміння ефективної організації структур даних та розробки алгоритмів їх обробки. Даний курс призначений для навчання студентів об'єктно-орієнтованому програмуванню і здобуття навиків у реалізації алгоритмів на мові програмування Java.

Метою навчальної дисципліни «Об'єктно-орієнтоване програмування» є забезпечення: отримання студентами теоретичних знань і практичних навичок програмування об'єктно-орієнтованою мовою програмування Java, засвоєння студентами принципів розробки програмних додатків з графічним інтерфейсом, формування навичок програмування засобами мови Java в інтегрованому середовищі NetBeans.

Загальні компетентності (ЗК):

ЗК 3. Здатність до абстрактного мислення, аналізу та синтезу.

ЗК 4. Здатність застосовувати знання у практичних ситуаціях.

ЗК 5. Знання та розуміння предметної області та розуміння професійної діяльності.

ЗК 8. Здатність вчитися і оволодівати сучасними знаннями.

Спеціальні компетентності (СК):

СК1. Здатність використовувати основні поняття, ідеї та методи фундаментальних наук під час розв'язання складних спеціалізованих задач з комп'ютерних наук в галузі інформаційних технологій.

СК2. Здатність використовувати теоретичні та фундаментальні знання в галузі комп'ютерних наук та інформаційних технологій для вирішення різноманітних проблем.

СК3. Здатність розробляти, аналізувати та застосовувати ефективні алгоритми для розв'язання конкретних професійних задач залежно від предметного середовища.

СК4. Здатність здійснювати проектування та розробку програмного забезпечення.

СК8. Здатність застосовувати сучасні методи, технології та інструментальні засоби проектування й створення програмних систем та їх супроводження.

Очікувані результати навчання

Результати навчання (РН):

РН03. Використовувати професійно-профільовані знання і практичні навички методів фундаментальної та прикладної математики під час розв'язання стандартних задач і задач прикладного характеру в галузі комп'ютерних наук.

РН04. Застосовувати сучасні методи математичного та комп'ютерного моделювання і будувати ефективні алгоритми для чисельного дослідження та розв'язання прикладних задач.

РН05. Розуміти основні методи і технології об'єктно-орієнтованого та компонентного програмування.

РН 10. Знати методології, методи, моделі, процеси і технології життєвого циклу розробки та тестування програмного забезпечення.

РН 12. Знати основні принципи функціонування системного та прикладного програмного забезпечення.

Передумовами вивчення даної дисципліни є вивчення шкільного курсу «Інформатика». Дисципліна «Об'єктно-орієнтоване програмування» є необхідною для подальшого вивчення дисциплін «Теорія алгоритмів», «Крос-платформне програмування», «Технологія створення програмних продуктів», «Технології захисту інформації» та ін.

В процесі викладання дисципліни використовується **5-бальна система оцінювання**.

У навчальному процесі використовуються такі методи навчання:

- лекція;
- лабораторна робота;
- самостійна робота;
- курсова робота;
- навчальна практика.

Оцінювання знань студентів здійснюється у формі усного опитування, тестів, оцінювання лабораторних робіт. Після вивчення курсу «Об'єктно-орієнтоване програмування» студенти складають екзамен.

Форми та засоби поточного і підсумкового контролю

Об'єктивність оцінки знань студентів залежить від форм та засобів поточного й підсумкового контролю. Процес вивчення курсу передбачає здійснення поточного та підсумкового контролю. Форми проведення поточного контролю, їх періоди визначаються робочим планом викладача. Поточний контроль проводиться викладачем на лекційних, контрольних, лабораторних заняттях, під час перевірки самостійних робіт. Формою підсумкового семестрового контролю є екзамен, що визначається навчальним планом. Підсумковий контроль здійснюється після вивчення курсу під час екзамену. До екзамену допускаються студенти, які засвоїли всі теми програми, виконали лабораторні роботи, рубіжну контрольну роботу.

КРИТЕРІЇ ОЦІНЮВАННЯ

(5-бальна шкала)

Критерії оцінки знань і вмінь

"відмінно" – «5» (високий рівень) – якщо студент вільно, глибоко й у повному обсязі засвоїв програмний матеріал; вичерпно, логічно викладає теоретичний матеріал в усній і письмовій формі; швидко і впевнено приймає правильні рішення при виконанні практичних завдань; має стійкі навички рішення різних задач;

"добре" – «4» (достатній рівень) – якщо студент вільно володіє матеріалом; у повному обсязі засвоїв програмний матеріал; здатний самостійно приймати правильні рішення при виконанні практичних завдань; має добрі навички рішення практичних задач;

"задовільно" – «3» (середній рівень) якщо студент може самостійно виконати більшу частину навчального матеріалу, дати основні поняття та визначення; допускає неточності при прийнятті рішень; при вирішенні практичних завдань потребує додаткових вказівок.

"незадовільно" – «2» (початковий рівень) – якщо студент засвоїв і може відтворити лише окремі питання програми; допускає суттєві помилки при прийнятті рішень; не має стійких навичок при вирішенні практичних завдань.

ЗМІСТ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Тема №1. Вступ. Мова програмування Java

План лекційного заняття №1:

1. Термінологія програмування
2. Класифікація мов програмування
3. Сучасні мови програмування
4. Об'єктно-орієнтоване програмування (ООП).
5. Мови ООП

Самостійна робота:

- ✓ Мови ООП, місце у сучасній ІТ-індустрії.

Перелік рекомендованих джерел інформації

Основний: 1, 2, 3.

Додатковий: 4, 5, 6, 7.

План лекційного заняття №2:

1. Мови ООП.
2. Мова програмування JAVA
3. Пакет JDK для розробника. Oracle - офіційний вендор JDK
4. Інтегровані середовища розробки (IDE).IDE NetBeans.

Самостійна робота:

- ✓ Мова програмування JAVA, місце у сучасній ІТ-індустрії.

Перелік рекомендованих джерел інформації

Основний: 1, 2, 3.

Додатковий: 4, 5, 6, 7, 8.

План лекційного заняття №3:

1. Перша програма мовою Java.
2. Компіляція та запуск програми.
3. Аналіз результату (тестування).
4. Виправлення помилок.

Самостійна робота:

- ✓ Етапи розробки програми.

Перелік рекомендованих джерел інформації

Основний: 1, 2, 3.

Додатковий: 4, 5, 6, 7, 9.

План лекційного заняття №4:

1. Типи даних. Розширювальні та звужувальні перетворення типів.
2. Обчислення математичних виразів засобами Java.
3. Типи алгоритмів: лінійний, розгалужений, циклічний.
4. Реалізація алгоритмів мовою Java

Самостійна робота:

- ✓ Типи алгоритмів.

Перелік рекомендованих джерел інформації

Основний: 1, 2, 3.

Додатковий: 4, 5, 6, 7, 10.

План лекційного заняття №5:

1. Масиви у Java.
2. Синтаксис масивів у Java.
3. Двовимірні масиви.
4. Приклади використання масивів.

Самостійна робота:

- ✓ Приклади використання масивів..

Перелік рекомендованих джерел інформації

Основний: 1, 2, 3.

Додатковий: 4, 5, 6, 7, 11.

ЛАБОРАТОРНА РОБОТА №1. Підготовка програмного забезпечення

Мета: *Встановити на ПК програмне забезпечення: JDK та NetBeans*

ЛАБОРАТОРНА РОБОТА №2. Етапи створення простої програми.

Мета: *Написати просту програму мовою Java, відкомпілювати та запустити програму, проаналізувати результат (тестування), виправити помилки.*

ЛАБОРАТОРНА РОБОТА №3. Структура проєкту в інтегрованому середовищі Netbeans

Мета: *навчитись створювати проєкт в інтегрованому середовищі Netbeans, засвоїти поняття «головний клас проєкту», «головний метод»; навчитись імпортувати класи, навчитись використовувати клас Scanner для читання даних, які поступають з клавіатури від користувача.*

ЛАБОРАТОРНА РОБОТА №4. Бібліотеки Util, Math

Мета: *засвоїти базові типи та математичні функції мови Java, навчитись підключати (імпортувати) бібліотеки Java, навчитись генерувати випадкові числа з допомогою математичної функції random, навчитись генерувати випадкові числа з допомогою класу Random.*

ЛАБОРАТОРНА РОБОТА №5. Лінійний алгоритм мовою Java

Мета: *навчитись будувати лінійний алгоритм мовою Java, організувати діалог з користувачем у консолі з допомогою класу Scanner, обчислювати математичні вирази, які містять математичні функції.*

ЛАБОРАТОРНА РОБОТА №6. Розгалужений алгоритм мовою Java

Мета: *навчитись будувати розгалужені алгоритми, використовуючи: умовний оператор if, скорочений умовний оператор if, оператор switch; навчитись складати складені умови з допомогою логічних операторів.*

ЛАБОРАТОРНА РОБОТА №7. Циклічний алгоритм мовою Java

Мета: навчитись будувати циклічний алгоритм, застосовувати арифметичний оператор *for*, оператори з умовою, вкладені цикли.

ЛАБОРАТОРНА РОБОТА №8. Масиви мовою Java. Одновимірні масиви.

Мета: засвоїти етапи створення масивів, розглянути задачі з використанням одновимірних масивів.

ЛАБОРАТОРНА РОБОТА №9. Масиви мовою Java. Двовимірні масиви.

Мета: засвоїти етапи створення масивів, розглянути задачі з використанням двовимірних масивів.

ЛАБОРАТОРНА РОБОТА №10. Масиви мовою Java.

Мета: розглянути задачі побудови ігрових полів з використанням двовимірних масивів; побудувати матрицю для гри «Морський бій», побудувати матрицю для гри «П'ятнашки», побудувати матрицю для гри «Сапер».

Тема №2. Об'єктно-орієнтоване програмування

План лекційного заняття №6:

1. Клас. Екземпляр класу.
2. Правила опису класу. Конструктори класу. Поля і методи класу.
3. Доступ до елементів класу. Специфікатори доступу до елементів (полів і методів) класу. Інкапсуляція.
4. Класи і підкласи.
5. Побудова проєкту з декількох класів.

Самостійна робота:

- ✓ Правила опису класу.

Перелік рекомендованих джерел інформації

Основний: 1, 2, 3.

Додатковий: 4, 5, 6, 7, 12.

План лекційного заняття №7:

1. Принципи ООП. Поліморфізм у Java: успадкування та перевизначення.
2. Абстрактні методи та класи.
3. Внутрішні та анонімні класи.
4. Знищення об'єктів та фіналізатори.
5. Виключення.

Самостійна робота:

- ✓ Інтерфейси.

Перелік рекомендованих джерел інформації

Основний: 1, 2, 3.

Додатковий: 4, 5, 6, 7, 13.

План лекційного заняття №8:

1. Пакет java.io. Система уведення/виведення. Потoki даних (stream).
2. Робота з файловою системою. Клас File.
3. Методи для створення, перейменування, видалення файлу, для запису даних у файл.
4. Класи-реалізації потоків даних.
5. Класи ByteArrayInputStream і ByteArrayOutputStream.
6. Класи FileInputStream і FileOutputStream.

Самостійна робота:

- ✓ Класи-реалізації потоків даних.

Перелік рекомендованих джерел інформації

Основний: 1, 2, 3.

Додатковий: 4, 5, 6, 7.

ЛАБОРАТОРНА РОБОТА №11. ООП. Правила опису класу.

Мета: засвоїти правила опису класу, побудувати клас, створити екземпляри класу та використати їх у програмі.

ЛАБОРАТОРНА РОБОТА №12. ООП. Рівні доступу до елементів класу. Інкапсуляція.

Мета: засвоїти правила опису класу, побудувати клас, створити екземпляри класу та використати їх у програмі за індивідуальним завданням.

ЛАБОРАТОРНА РОБОТА №13. ООП. Ієрархія класів

Мета: Організація проекту, який містить декілька класів (індивідуальне завдання)

ЛАБОРАТОРНА РОБОТА №14. ООП. Поліморфізм: успадкування, перевизначення.

Мета: розглянути приклади побудови класів-нащадків, засвоїти правила успадкування та перевизначення елементів класів

ЛАБОРАТОРНА РОБОТА №15. ООП. Клас File

Мета: Клас File. Створення, перейменування, видалення файлу, запис даних у файл. Провести рубіжний контроль.

Тема №3. Розробка графічного інтерфейсу програмного додатку засобами GUI Builder

План лекційного заняття №9:

1. Методи створення компонентів графічного інтерфейсу програмного додатку.
2. Елементи управління.
3. Ієрархія класів-подій.
4. Обробка подій.

Самостійна робота:

- ✓ Ієрархія класів-подій.

Перелік рекомендованих джерел інформації

Основний: 1, 2, 3.

Додатковий: 4, 5, 6, 7, 8.

План лекційного заняття №10:

1. Розробка та збирання компонентів в середовищі GUI Builder. Об'єкти та сервіси, що ним надаються.

2. Створення компонентів типу Windows Forms.

3. Компонети: JLabel, JTextField, JButton, JPanel та інші. Властивості компонентів.

4. Розробка дизайну програмного додатку.

Самостійна робота:

- ✓ Властивості компонентів: JLabel, JTextField, JButton, JPanel.

Перелік рекомендованих джерел інформації

Основний: 1, 2, 3.

Додатковий: 4, 5, 6, 7, 9.

ЛАБОРАТОРНА РОБОТА №16. Організація проєкту

для створення компонентів типу Windows Forms в середовищі GUI Builder

Мета: Створити Frame Form в середовищі GUI Builder.

ЛАБОРАТОРНА РОБОТА №17. Розробка компонентів типу

Windows Forms в середовищі GUI Builder

Мета: Створити Frame Form в середовищі GUI Builder, розробити додаток Каса, навчитись використовувати клас Formatter для форматування числа, навчитись використовувати методи перетворення тексту у число.

ЛАБОРАТОРНА РОБОТА №18. Імпортування проєкту

Мета: Створити Frame Form в середовищі GUI Builder, розробити додаток Калькулятор, навчитись упаковувати проєкт шляхом імпортування у формат ZIP.

ЛАБОРАТОРНА РОБОТА №19. Імпортування проєкту

Мета: Створити Frame Form в середовищі GUI Builder, розробити додаток для редагування тексту, використати клас String та клас StringBuffer.

ЛАБОРАТОРНА РОБОТА №20. Розробка компонентів типу

Windows Forms. Підготовка до курсової роботи

Мета: розглянути додаток, який будує графік функції.

Тема №4. Розробка графічного інтерфейсу програмного додатку з використанням менеджерів компоновки

План лекційного заняття №11:

1. Менеджери компоновки елементів графічного інтерфейсу програмного додатку.
2. Менеджери компоновки: FlowLayout, GridLayout, BoxLayout, BorderLayout, CardLayout, GridBagLayout, GroupLayout.
3. Використання менеджера компоновки GridLayout на прикладі гри Сапер.

Самостійна робота:

- ✓ Менеджери компоновки.

Перелік рекомендованих джерел інформації

Основний: 1, 2, 3.

Додатковий: 4, 5, 6, 7, 10.

План лекційного заняття №12:

1. Використання менеджерів компоновки на прикладі гри П'ятнашки.
2. Гейм-дизайн.
3. Розміщення файлів типу Jpeg на елементах графічного інтерфейсу додатку.

Самостійна робота:

- ✓ Менеджери компоновки.

Перелік рекомендованих джерел інформації

Основний: 1, 2, 3.

Додатковий: 4, 5, 6, 7, 11.

План лекційного заняття №13:

1. Графічні примітиви у Java.
2. Класи Graphics і Graphics2D.
3. Приклади використання класів Graphics і Graphics2D у гейм-дизайні програмних додатків.

Самостійна робота:

- ✓ Класи Graphics і Graphics2D.

Перелік рекомендованих джерел інформації

Основний: 1, 2, 3.

Додатковий: 4, 5, 6, 7, 12.

План лекційного заняття №14:

1. Аплет.
2. Розробка аплету у інтегрованому середовищі NetBeans.
3. Приклад розробки аплету.

Самостійна робота:

- ✓ Аплет.

Перелік рекомендованих джерел інформації

Основний: 1, 2, 3.

Додатковий: 4, 5, 6, 7, 13.

План лекційного заняття №15:

1. Перспективи об'єктно-орієнтованого програмування
2. Перспективи використання мови Java.

Самостійна робота:

- ✓ Перспективи використання мови Java.

Перелік рекомендованих джерел інформації

Основний: 1, 2, 3.

Додатковий: 4, 5, 6, 7.

ЛАБОРАТОРНА РОБОТА №21. Менеджери компоновки.

Мета: дослідити програмний код гри *Сапер*, пройти тестування.

ЛАБОРАТОРНА РОБОТА №22. Менеджер компоновки GridLayout

Мета: змінити дизайн комп'ютерної гри *Сапер*, відредагувавши програмний код гри; використати методи *set* для форматування шрифту, кольору, розміру елементів графічного інтерфейсу програми.

ЛАБОРАТОРНА РОБОТА №23. Менеджер компоновки GridLayout

Мета: дослідити програмний код гри *П'ятнашки*, пройти тестування.

ЛАБОРАТОРНА РОБОТА №24. Менеджер компоновки BorderLayout

Мета: змінити дизайн комп'ютерної гри *П'ятнашки*, відредагувавши програмний код гри; використати методи *set* для форматування шрифту, кольору, розміру елементів графічного інтерфейсу програми.

ЛАБОРАТОРНА РОБОТА №25. Розробка компонентів типу Windows Forms. Менеджери компоновки.

Мета: розробити структуру ігрового додатку *Палички* з графічним інтерфейсом.

ЛАБОРАТОРНА РОБОТА №26. Розробка компонентів типу Windows Forms. Класи Graphics і Graphics2D

Мета: розробити дизайн ігрового додатку *Палички* з графічним інтерфейсом.

ЛАБОРАТОРНА РОБОТА №27. Контрольна робота

Мета: пройти тестування.

ЛАБОРАТОРНА РОБОТА №28. Підведення підсумків

Мета: підведення підсумків.

ПЕРЕЛІК ПИТАНЬ ДЛЯ РУБІЖНОГО КОНТРОЛЮ з курсу «Об'єктно-орієнтоване програмування»

1. Причини переходу від процедурного до об'єктно-орієнтованого програмування.
2. Описати переваги та недоліки об'єктно-орієнтованого програмування.
3. Правила опису класів. Створення об'єктів класу. Особливості конструкторів у Java.
4. Поліморфізм у Java. Розкрийте сутність поняття поліморфізм.
5. Що таке інкапсуляція і поліморфізм. Описати типи відносин між класами.
6. Модифікатори доступу. Які модифікатори дозволяють звертатися до елемента з класів того ж пакету?
7. Абстрактні методи та класи. Інтерфейси.
8. Знищення об'єктів та фіналізатори.
9. Виключення.
10. Внутрішні та анонімні класи.
11. Описати використання статичних елементів.
12. Розкрийте застосування ключових слів `this` і `super`.
13. Розкрийте методи роботи з інтерфейсами в Java. Опишіть їх.
14. Розкрийте поняття імена і пакети в Java. Розкрийте область видимості імен в Java.
15. Платформи, програмні моделі та інструментальні засоби для створення Java-застосувань.
16. Пакети в Java.
17. Основні механізми ініціалізації.
18. Особливості рядків у Java. Операції над рядками.
19. Запуск і базові властивості потоків. Синхронізація.
20. Як записати в Java-програмі символ з кодом 6915?
21. Які види лексем є в Java? Опишіть їх.
22. Типи даних в Java. Примітивні і посилальні типи даних. Різниця між ними.
23. Наведіть приклад значення цілочисельний змінної `x`, при якій такі вирази не вірні.
24. Розкрийте процес перетворення типів в Java.
25. Розкрийте засоби створення та використання масивів в Java.
26. Розкрийте методи застосування умовних та циклічних операторів в Java.
27. Помилки при роботі програми. Винятки в Java.
28. Масиви та списки. Сортування масивів.
29. Додавання та видалення елементів масиву.
30. Змінні і масиви. Взаємні переходи.

ПЕРЕЛІК ПИТАНЬ ДО ЕКЗАМЕНУ

1. Принципи реалізації платформної незалежності.
2. Проблема захисту. Модель пісочниці.

3. Платформи, програмні моделі та інструментальні засоби для створення Java-застосувань.
4. Процедурні аспекти програмування: основні особливості.
5. Особливості рядків у Java.
6. Пакети в Java.
7. Створення об'єктів. Особливості конструкторів у Java.
8. Модифікатори доступу.
9. Основні механізми ініціалізації.
10. Поліморфізм у Java.
11. Абстрактні методи та класи. Інтерфейси.
12. Знищення об'єктів та фіналізатори.
13. Виключення.
14. Внутрішні та анонімні класи.
15. Колекції та сортування колекцій.
16. Сокети.
17. Запуск і базові властивості потоків. Синхронізація.
18. Графічні застосування. Модель обробки подій.
19. Проаналізувати причини переходу з процедурного до об'єктно-орієнтованого програмування.
20. Як записати в Java-програмі символ з кодом 6915?
21. Описати переваги та недоліки об'єктно-орієнтованого програмування.
22. Що таке інкапсуляція і поліморфізм. Описати типи відносин між класами.
23. Які модифікатори дозволяють звертатися до елемента з класів того ж пакету?
24. Спроектуйте клас «Кавовий автомат». Напишіть, які властивості і методи ви б включили в цей клас.
25. Які види лексем є в Java? Опишіть їх.
26. Чому буде дорівнювати наступний вираз і значення змінних x і y після обчислень?
27. Типи даних в Java. Примітивні і посилальні типи даних. Різниця між ними.
28. Наведіть приклад значення цілочисельний змінної x, при якій такі вирази не вірні.
29. Розкрийте поняття імена і пакети в Java.
30. Розкрийте область видимості імен в Java.
31. Розкрийте процес перетворення типів в Java.
32. Описати використання статичних елементів.
33. Розкрийте застосування ключових слів this і super.
34. Розкрийте методи роботи з інтерфейсами в Java. Опишіть їх.
35. Розкрийте сутність поняття поліморфізм.
36. Розкрийте засоби створення та використання масивів в Java.
37. Розкрийте методи застосування умовних та циклічних операторів в Java.
38. Помилки при роботі програми. Винятки в Java.

39. Масиви та списки.
40. Сортування масивів.
41. Додавання та видалення елементів масиву.
42. Змінні і масиви. Взаємні переходи.
43. Операції над рядками.
44. Які графічні примітиви можна реалізувати у Java?
45. Які є візуальні бібліотеки у Java?
46. Назвіть призначення Класів Graphics і Graphics2D.
47. Назвіть методи графічних примітивів.
48. Назвіть елементи компонентів типу Windows Forms.
49. Які події Ви знаєте? Як реалізовується обробка подій у Java?
50. Що таке аплет?

СПИСОК РЕКОМЕНДОВАНИХ ДЖЕРЕЛ ІНФОРМАЦІЇ

Основна:

1. Горбань А. Г. Програмування в Java, 2008. 310 с.
2. Ратушняк Т. В. Програмування мовою Java : практикум / Т.В. Ратушняк– Ірпінь : Університет ДФС України, 2017. 214 с.
3. Java-програмування: комп'ютерний практикум [Електронний ресурс] : навч. посіб. для студ. спеціальності 122 «Комп'ютерні науки», освітньо-професійної програми «Комп'ютерний моніторинг та геометричне моделювання процесів і систем» / КПІ ім. Ігоря Сікорського ; уклад.: Ю.А. Тарнавський. – Електронні текстові дані (1 файл: 686 Кбайт). – Київ : КПІ ім. Ігоря Сікорського, 2021. – 95 с.

Допоміжна:

4. Васильєв О.М. Програмування мовою Java / О.М. Васильєв. — Тернопіль: Навчальна книга – Богдан, 2019. 36 с.
5. Кадомський К.К., Ніколюк П.К. Java. Теорія і практика: навчальний посібник для студентів природничих спеціальностей університетів / К.К. Кадомський, П.К. Ніколюк. – Вінниця: ДонНУ, 2019. – 197 с.
6. Об'єктно орієнтоване програмування на Java. Конспект лекцій з дисципліни «Об'єктно орієнтоване програмування» для студентів спеціальностей 121 – «Програмна інженерія», 123 – «Комп'ютерна інженерія». / Укл.: Бивойно П.Г. – Чернігів: ЧНТУ, 2019. – 136 с.
7. Яценков В.С. Java за тиждень. — Видавнича система: Ridero, 2018. 312 с.

Інформаційні ресурси Інтернет:

8. Java™ Platform, Enterprise Edition 6 API Specification. — <http://docs.oracle.com/javaee/6/api/>

9. The Java Tutorials. – <http://docs.oracle.com/javase/tutorial/>
10. Інформація про курс JAVA101 | Prometheus. – https://edx.prometheus.org.ua/courses/EPAM/JAVA101/2016_T2/info
11. Комп'ютерне програмування (Електронний навчальний курс). – <http://it.nubip.edu.ua/course/view.php?id=150>
12. Основи Java. - http://iwanoff.inf.ua/java_ua/index.html
13. Netbeans Apache. - <https://netbeans.apache.org>