

Міністерство освіти і науки України
Харківський національний університет імені В.Н. Каразіна
Кафедра штучного інтелекту та програмного забезпечення



РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Крос-платформне програмування

рівень вищої освіти	перший (бакалаврський) рівень
галузь знань	12 Інформаційні технології
спеціальність	122 Комп'ютерні науки
освітня програма	Комп'ютерні науки
галузь знань	12 Інформаційні технології
спеціальність	123 Комп'ютерна інженерія
освітня програма	Комп'ютерна інженерія
галузь знань	12 Інформаційні технології
спеціальність	125 Кібербезпека
освітня програма	Кібербезпека
галузь знань	15 Автоматизація та приладобудування
спеціальність	151 Автоматизація та комп'ютерно-інтегровані технології
освітня програма	Автоматизація та комп'ютерно-інтегровані технології
вид дисципліни	обов'язкова
факультет	комп'ютерних наук


Програму рекомендовано до затвердження Вченою радою факультету комп'ютерних наук
«29» серпня 2018 року, протокол № 9

РОЗРОБНИКИ ПРОГРАМИ:

кандидат фізико-математичних наук, доцент кафедри штучного інтелекту та програмного
забезпечення **Споров Олександр Євгенович**

Програму схвалено на засіданні кафедри штучного інтелекту та програмного забезпечення
Протокол від «26» червня 2018 року № 11

Завідувач кафедри штучного інтелекту та програмного забезпечення


_____ (Куклін В.М.)

Програму погоджено методичною комісією факультету комп'ютерних наук
Протокол від «27» червня 2018 року № 7

Голова методичної комісії факультету комп'ютерних наук


_____ (Васильєва Л.В.)



ВСТУП

Програма навчальної дисципліни «Крос-платформне програмування» складена відповідно до освітньо-професійної програми підготовки (бакалаврського) рівня вищої освіти, напрямів підготовки: 122 «Комп'ютерні науки та інформаційні технології», 125 «Кібербезпека», 151 «Автоматизація та комп'ютерно-інтегровані технології», 123 «Комп'ютерна інженерія»

1. Опис навчальної дисципліни

1.1. Метою навчальної дисципліни є засвоєння студентами як фундаментальних принципів компонентної та розподіленої організації програм, так і прийомів практичного створення компонентних розподілених програмних продуктів на широко розповсюдженій мові крос-платформного програмування *Java*.

1.2. Основними завданнями вивчення дисципліни є:

- ознайомлення з архітектурами та стандартами компонентних моделей;
- ознайомлення з принципами проектування та розробки віддалених компонентів;
- ознайомлення з основами взаємодії різних програмних платформ в розподілених корпоративних інформаційних системах;
- ознайомлення з комунікаційними засобами розподілених обчислень;
- отримання умінь розв'язувати проблеми масштабованості.

1.3. Кількість кредитів – 4

1.4. Загальна кількість годин - 120

1.5. Характеристика навчальної дисципліни	
<u>Нормативна / за вибором</u>	
денна форма навчання	Заочна (дистанційна) форма навчання
Рік підготовки	
2-й	
Семестр	
4-й	
Лекції	
32 год.	___ год.
Практичні, семінарські заняття	
___ год.	___ год.
Лабораторні заняття	
32 год.	___ год.
Самостійна робота	
21 год.	___ год.
Індивідуальні завдання	
35 год.	___ год.

1.6. Заплановані результати навчання

знати:

- основні технології, що лежать в основі та забезпечують можливість організації компонентної та розподіленої архітектури комп'ютерних програм;
- основи технології компонентного програмування та технології створення розподілених програм різної архітектури.

вміти:

- проектувати та створювати компонентні та розподілені програмні комплекси за допомогою широко розповсюдженій мові крос-платформного програмування *Java*.

2. Тематичний план навчальної дисципліни

Розділ 1. Самоаналіз.

Тема 1. Рефлексія.

Поняття рефлексії. Клас `Class`. Аналіз властивостей класів. Аналіз конструкторів класу. Аналіз методів класу. Аналіз полів класу.

Тема 2. Рефлексивні «показчики» на методи. Класи — посередники.

Об'єкт `Method`. Перелік аргументів методу. Рефлексивне створення нових об'єктів та масивів. Завантажувач класів. Поняття про `Proxy-classes`. Обробник викликів. Властивості та використання класів-посередників.

Розділ 2. Персистентність.

Тема 1. Стандартний інтерфейс серіалізації `Serializable`.

Поток введення-виведення. Об'єктні потоки. Стандартний інтерфейс серіалізації. Посилання на об'єкти, що не серіалізуються. `Transient` поля. Версії серіалізації.

Тема 2. Розширений інтерфейс серіалізації `Externalizable`.

Розробка формату серіалізації. Методи `readObject` та `writeObject`. Методи `defaultReadObject` та `defaultWriteObject`. Розширений інтерфейс серіалізації.

Тема 3. Стандарт XML. Схема DTD та SXD. Зв'язок документа та схеми.

Стандарт XML. Базові елементи XML документа. Коректні XML документи. Схеми XML документів. Зв'язок документа та схеми.

Тема 4. Обробка XML документа. SAX – парсери.

Різні підходи до обробки XML документів. `Simple API for XML`. Основні властивості; переваги та недоліки. Базові повідомлення. Обробники повідомлень. Валідуючі та не валідуючі SAX – парсери.

Тема 5. Модель DOM. DOM парсери.

Специфікація `Document Object Model`. Стандарти DOM. Модель документа. Побудова дерева документа. Класи, методи та атрибути цих методів, що призначені для аналізу структури документів та роботи із представленням документів у вигляді дерева. Валідуючі та не валідуючі DOM – парсери. Основні переваги валідуючих DOM – парсерів. Збереження XML документів.

Розділ 3. Компонентне та мережеве програмування.

Тема 1. Компонентний підхід до створення програмних продуктів.

Основні технології програмування. Компонентно орієнтоване програмування як розвиток об'єктно орієнтованого програмування. Компонент. Властивості компонента. Визначення компонентів. Специфікація інтерфейсу як контракту. Основні вимоги до компонентного програмного забезпечення.

Тема 2. Компонентна технологія Java Beans.

Основні поняття компонентної ідеології Java Beans. Компоненти Java Beans. Контейнери Java Beans. Властивості компонентів: скалярні, індексовані та булеві. Пов'язані та обмежені властивості. Події компонентів. Персистентність компонентів. Особливості налаштування та серіалізації компонентів. Серіалізація в XML файли.

Тема 3. Мережеве програмування. Сокети.

Мережа Інтернет. Протокол. Стек протоколів TCP/IP. Класи для представлення адреси. Протокол TCP/IP. Сокети TCP/IP. Базові правила роботи з потоковими сокетами. Основні переваги та недоліки. Передача об'єктів за допомогою сокетів через мережу. TCP/IP сервери та клієнти. Однопоточний та багатопоточний сервер. Розробка власного протоколу віддаленого виклику методів.

Тема 4. Мережеве програмування. Датаграми.

Датаграми. Основні переваги та недоліки датаграм. Класи, що представляють датаграми. Передача об'єктів та масивів об'єктів за допомогою датаграм. Датаграм-сервер. Датаграм-клієнт.

Тема 5. Мережеве програмування. Багатоадресна передача інформації.

Особливості роботи з великою кількістю клієнтів. Можливість організації багатоадресної (multicast) передачі інформації. Класи, що надають можливість одночасної роботи з декількома клієнтами. Особливості багатоадресної (multicast) передачі інформації.

Розділ 4. Розподілене програмування.

Тема 1. Розподілена архітектура компонентних систем. Технологія Java RMI.

Основні ознаки розподіленої системи. Фактори розвитку розподілених систем. Переваги та недоліки розподілених систем. Основні вимоги до розподілених систем. Проміжне середовище. Технологія розподіленого програмування Java RMI. Віддалені об'єкти. Основні інтерфейси та класи для створення розподілених доданків за технологією Java RMI. Структура найпростішого розподіленого доданку.

Тема 2. Розгортання розподілених доданків Java RMI.

Віддалений виклик методу. Віддалені інтерфейси. Віддалена передача параметрів: заглушки та скелетони. Використання динамічних гроху. Служба реєстру RMI. Створення / запуск служби реєстру. Реєстація віддалених об'єктів в службі реєстру. Динамічне завантаження класів. Розгортання віддалених доданків на декількох вузлах мережі.

Тема 3. Бази даних, як невід'ємна складова розподілених систем.

Бази даних – важливий компонент розподілених систем. Дво- та трьох-ланкові архітектури. Сервери баз даних. Простий стандартних сервер Java DB (Derby). Створення простої бази даних. Основні засоби мови SQL.

Тема 4. Основи JDBC.

Основні принципи роботи з базами даних в Java. Основні класи та інтерфейси для взаємодії з базами даних. Створення автономного та розподіленого доданків, що взаємодіють з сервером бази даних Java DB.

3. Структура навчальної дисципліни

Назви розділів і тем	Кількість годин					
	денна форма					
	усього	у тому числі				
л		п	лаб.	інд.	с. р.	
1	2	3	4	5	6	7
Розділ 1. Самоаналіз						
Тема 1. Рефлексія.	6	2		2	1	1
Тема 2. Рефлексивні «показчики» на методи. Класи-посередники.	8	2		2	3	1
Разом за розділом 1	14	4		4	4	2
Розділ 2. Персистентність						
Тема 1 Стандартний інтерфейс серіалізації Serializable.	6	2		2	1	1
Тема 2. Розширений інтерфейс серіалізації Externalizable.	7	2		2	2	1
Тема 3. Стандарт XML. Схема DTD та SXD. Зв'язок документа та схеми.	7	2		2	2	1
Тема 4. Обробка XML документа. SAX – парсери.	8	2		2	3	1
Тема 5. Модель DOM. DOM парсери.	8	2		2	3	1
Разом за розділом 2	36	10		10	11	5
Розділ 3. Компонентне та мережеве програмування						
Тема 1. Компонентний підхід до створення програмних продуктів.	6	2		2	1	1
Тема 2. Компонентна технологія <i>Java Beans</i> .	8	2		2	2	2
Тема 3. Мережеве програмування. Сокети.	8	2		2	3	1
Тема 4. Мережеве програмування. Датаграми.	8	2		2	3	1
Тема 5. Мережеве програмування. Multicast sockets.	8	2		2	3	1
Разом за розділом 3	38	10		10	12	6
Розділ 4. Розподілене програмування						
Тема 1. Розподілена архітектура компонентних систем.	8	2		2	2	2
Тема 2. Розгортання розподілених доданків Java RMI.	8	2		2	2	2
Тема 3. Бази даних, як невід'ємна складова розподілених	8	2		2	2	2
Тема 4. Основи JDBC	8	2		2	2	2
Разом за розділом 4	32	8		8	8	8
Усього годин	120	32		32	35	21

4. Теми практичних (лабораторних) занять

№ з/п	Назва теми	Кількість годин
Розділ 1. Самоаналіз		
1	Об'єктно-орієнтоване програмування: система роботи з функціями	2
2	Рефлексивний аналіз класів	2
3	Створення об'єктів та масиву об'єктів за допомогою рефлексії. Рефлексивні «показчики» на методи. Динамічні Proxy-класи	2
Розділ 2. Персистентність		
4	Серіалізація системи за допомогою інтерфейсу Serializable	2
5	Серіалізація системи за допомогою інтерфейсу Externalizable	2
6	Створення XML документів та схем, зв'язок документа та схеми	2
7	Обробка XML документів за допомогою SAX парсеру	2
8	Обробка XML документів за допомогою DOM парсеру. Збереження документу.	2
Розділ 3. Компонентне та мережеве програмування		
9	Створення візуальних компонентів для зберігання та для графічного відображення даних	2
10	Створення програми з візуальним інтерфейсом користувача з стандартних та власних компонентів	2
11	TCP сокети. Мережева клієнт-серверна програма обробки інформації	2
12	Датаграми. Мережева клієнт-серверна програма передачі інформації	2
13	Багатоадресні сокети. Мережева клієнт-серверна програма багатоадресної передачі інформації	2
Розділ 4. Розподілене програмування		
14	Проектування розподіленої обчислювальної системи. Розробка віддалених серверних та клієнтських інтерфейсів. Створення клієнтської частини RMI обчислювальної програми	2
15	Створення серверної частини RMI обчислювальної програми. Запуск розподіленої програми з можливістю динамічної завантаження файлів, що відсутні	2
16	Основи JDBC. Створення розподіленої системи, що взаємодіє з сервером бази даних.	2
Усього годин		32

5. Завдання для самостійної роботи

№ з/п	Види, зміст самостійної роботи.	Кількість годин
Розділ 1. Самоаналіз		
1	Рефлексія: аналіз класів. Самостійне опрацювання навчального матеріалу, що було надано під час лекційного заняття по Темі 1 із Розділу 1. Контроль: опитування під час практичного заняття, опитування під час здачі індивідуального розрахункового завдання, теоретичні питання в контрольній роботі № 1	1
2	Рефлексія: створення об'єктів, масивів, динамічних проксі. Узагальнений клас Class. Самостійне опрацювання навчального матеріалу, що було надано під час лекційного заняття по Темі 2 із Розділу 1. Контроль: опитування під час практичного заняття, опитування під час здачі індивідуального розрахункового завдання, теоретичні питання в контрольній роботі № 1	1
Разом		2
Розділ 2. Персистентність		
3	Серіалізація: основні поняття, інтерфейс Serializable. Використання серіалізації при створенні розподіленого програмного забезпечення. Самостійне опрацювання навчального матеріалу, що було надано під час лекційного заняття по Темі 1 із Розділу 2. Контроль: опитування під час практичного заняття, опитування під час здачі індивідуального розрахункового завдання, теоретичні питання в контрольній роботі № 1	1
4	Серіалізація. Розширений інтерфейс серіалізації Externalizable. Самостійне опрацювання навчального матеріалу, що було надано під час лекційного заняття по Темі 2 із Розділу 2. Контроль: опитування під час практичного заняття, опитування під час здачі індивідуального розрахункового завдання, теоретичні питання в контрольній роботі № 1	1
5	XML документи. DTD та XSD схеми XML документів. Java API для обробки XML документів (JAXP) . Самостійне опрацювання навчального матеріалу, що було надано під час лекційного заняття по Темі 3 із Розділу 2. Контроль: опитування під час практичного заняття, опитування під час здачі індивідуального розрахункового завдання, теоретичні питання в контрольній роботі № 1	1
6	Обробка XML документів за допомогою SAX парсеру. Самостійне опрацювання навчального матеріалу, що було надано під час лекційного заняття по Темі 4 із Розділу 2. Контроль: опитування під час практичного заняття, опитування під час здачі індивідуального розрахункового завдання, теоретичні питання в контрольній роботі № 1	1
7	Document Object Model. Обробка XML документів за допомогою DOM парсеру. Змінення DOM об'єкту. Трансформація DOM об'єкту та збереження його у файлі. Самостійне опрацювання навчального матеріалу, що було надано під час лекційного заняття по Темі 5 із Розділу 2. Контроль: опитування під час практичного заняття, опитування під час здачі індивідуального розрахункового завдання, теоретичні питання в контрольній роботі № 1	1
Разом		5
Розділ 3. Компонентне та мережеве програмування		
8	Компонентний підхід до створення програмних продуктів. Сучасні технології компонентно-орієнтованого програмування. Самостійне опрацювання навчального матеріалу, що було надано під час лекційного заняття по Темі 1 із Розділу 3. Контроль: опитування під час практичного заняття, опитування під	1

	час здачі індивідуального розрахункового завдання.	
9	Компонентна технологія <i>JavaBeans</i> . Робота з властивостями та подіями. Способи забезпечення персистентності компонентів. Налаштування компонентів: створення та підключення редакторів та настроювачів компонентів. Інтегрування компонентів користувача із основними середовищами розробки. Самостійне опрацювання навчального матеріалу, що було надано під час лекційного заняття по Темі 2 із Розділу 3. Контроль: опитування під час практичного заняття, опитування під час здачі індивідуального розрахункового завдання.	2
10	Клієнт-серверна архітектура. Особливості реалізації. Особливості TCP протоколу. Використання потокових (TCP) сокетів <i>Java</i> для створення мережесих клієнт / серверних застосунків. Самостійне опрацювання навчального матеріалу, що було надано під час лекційного заняття по Темі 3 із Розділу 3. Контроль: опитування під час практичного заняття, опитування під час здачі індивідуального розрахункового завдання.	1
11	Особливості протоколу UDP. Використання Datagram сокетів <i>Java</i> для створення мережесих клієнт / серверних застосунків. Самостійне опрацювання навчального матеріалу, що було надано під час лекційного заняття по Темі 4 із Розділу 3. Контроль: опитування під час практичного заняття, опитування під час здачі індивідуального розрахункового завдання.	1
12	Особливості багатоадресної передачі інформації в мережі. Використання Multicast сокетів <i>Java</i> для створення мережесих застосунків. Самостійне опрацювання навчального матеріалу, що було надано під час лекційного заняття по Темі 5 із Розділу 3. Контроль: опитування під час практичного заняття, опитування під час здачі індивідуального розрахункового завдання.	1
	Разом	6
Розділ 4. Розподілене програмування		
13	Основні сучасні технології розробки розподіленого програмного забезпечення. Основні особливості використання технології <i>Java RMI</i> для створення мережесих клієнт / серверних застосунків. Самостійне опрацювання навчального матеріалу, що було надано під час лекційного заняття по Темі 1 із Розділу 4. Контроль: опитування під час практичного заняття, опитування під час здачі індивідуального розрахункового завдання .	2
14	Служба RMI реєстру. Особливості розгортання RMI застосунків. Кодова база. Файли політики безпеки. Служба активації RMI об'єктів. Самостійне опрацювання навчального матеріалу, що було надано під час лекційного заняття по Темі 2 із Розділу 4. Контроль: опитування під час практичного заняття, опитування під час здачі індивідуального розрахункового завдання .	2
15	Бази даних, як невід'ємна складова розподілених систем. База даних <i>Java DB</i> . Проектування та створення простих баз даних. Основні засоби мови SQL. Самостійне опрацювання навчального матеріалу, що було надано під час лекційного заняття по Темі 3 із Розділу 4. Контроль: опитування під час практичного заняття, опитування під час здачі індивідуального розрахункового завдання .	2
16	Основи JDBC: основні інтерфейси та класи. Використання бази даних <i>Java DB</i> в за <i>Java</i> стосунках. Особливості обробки помилок. Самостійне опрацювання навчального матеріалу, що було надано під час лекційного заняття по Темі 4 із Розділу 4. Контроль: опитування під час практичного заняття, опитування під час здачі індивідуального розрахункового завдання .	2
	Разом	8
	Усього годин	27

6. Індивідуальні завдання

Загалом передбачено 2 обов'язкових розрахункових завдання, що виконуються студентами індивідуально. Для вибору, за наступними темами:

1. Розширення об'єктно орієнтованої системи диференціювання функцій: додавання нових можливостей до готової системи (нові власні функції; функції, що задані аналітично; графічне відображення функцій та похідних; створення графічного інтерфейсу користувача).

2. Рефлексивний аналіз класів і об'єктів, створення об'єктів та робота з такими об'єктами; рефлексивне створення масивів; динамічні проксі.

3. Розширена серіалізація / десеріалізація складної системи, частина класів якої не підтримує серіалізацію.

4. Створення дерева об'єктів при отриманні інформації з XML документу за допомогою валідуючого SAX парсеру, створення відповідного DOM - об'єкту, змінення його та збереження в новому XML-файлі.

5. Розширення функціональності власних *Java Beans* компонентів: можливість бінарної серіалізації / десеріалізації та серіалізації / десеріалізації в XML файл; створення власних редакторів властивостей.

6. Створення застосунку за допомогою *Java Beans* компонентів. Створення власних налаштувачів та редакторів властивостей *JavaBeans* компонентів.

7. Розширення функціональності мережевої програми обробки інформації, що базується на TCP сокетах. Створення клієнт / серверного застосунку з власним протоколом, що реалізує найпростіший варіант віддаленого виклику методів.

8. Створення мережевого застосунку з графічним інтерфейсом користувача, що реалізує багатоадресне спілкування.

9. Створення клієнт / серверного застосунку з графічним інтерфейсом користувача, що згідно з технологією *Java RMI* виконує реєстрацію учасників конференції на сервері конференції.

10. Розширення функціональних можливостей застосунку з Завдання № 9: робота з сервером бази даних *Java DB*.

7. Методи контролю

Контроль засвоєння навчального матеріалу здійснюється за допомогою:

- прийому та оцінювання завдань лабораторних робіт;
- прийому та оцінювання завдань індивідуальних обчислювальних робіт;
- проведення контрольної роботи за результатами відпрацювання основних положень навчальної програми.

8. Схема нарахування балів

Розподіл балів для підсумкового семестрового контролю при проведенні залікової роботи

Поточний контроль, самостійна робота, індивідуальні завдання																Контрольна	Самостійна робота	Індивідуальні завдання	Разом	Залікова робота	Сума
Розділ 1		Розділ 2					Розділ 3					Розділ 4									
T1	T2	T1	T2	T3	T4	T5	T1	T2	T3	T4	T5	T1	T2	T3	T4						
2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	6	8	14	60	40	100

Шкала оцінювання

Сума балів за всі види навчальної діяльності протягом семестру	Оцінка	
	для чотирирівневої шкали оцінювання	для дворівневої шкали оцінювання
90 – 100	відмінно	зараховано
70-89	добре	
50-69	задовільно	
1-49	незадовільно	не зараховано

Критерії оцінювання

Оцінка		Пояснення	
90-100	Відмінно	Зараховано	Теоретичний матеріал курсу засвоєно повністю, показано знайомство з додатковою інформацією, отримані всі необхідні навички практичної роботи. Студент розуміє взаємозв'язки розглянутих процесів. Всі практичні та самостійні завдання виконано в повному обсязі без помилок або з невеликими недоліками. Студент може логічно, послідовно, грамотно, точно, використовуючи спеціальні терміни, пояснити своє рішення та доказати його правильність. При цьому відсутні помилки та неточні формулювання. Показано вміння самостійно розв'язувати проблеми та робити правильні висновки з результатів.
70-89	Добре		Теоретичний матеріал курсу в основному засвоєно, отримані всі основні навички практичної роботи. Студент розуміє базові взаємозв'язки розглянутих процесів. Всі практичні та самостійні завдання виконано в повному обсязі з невеликою кількістю помилок або недоліків. Студент може, використовуючи спеціальні терміни, пояснити своє рішення та показати, що воно є вірним. При цьому допустимі деякі неточності в формулюванні або невеликі похибки в міркуваннях. Показано вміння самостійно розв'язувати проблеми.
50-69	Задовільно		Теоретичний матеріал курсу засвоєно не повністю, але те, що не засвоєно має не істотний характер. Базові навички практичної роботи в основному засвоєно. Студент розуміє розглянуті процеси. Більшість практичних та самостійних завдань виконано, можливо з помилками або недоліками. Студент може пояснити особливості свого рішення. Показано вміння знаходити путі розв'язку проблем.
1-49	Не задовільно		Не зараховано

9. Рекомендована література

Основна література

1. Хорстманн К., Корнелл Г. Java 2. Библиотека профессионала. том. 2. Тонкости программирования. 9-е изд. - М: «Вильямс», 2014 - 1008 с.
2. Нимейер П., Леук Д. Программирование на Java. - М.: «Эксмо», 2014. – 1216 с.

3. Эккель Б. Философия Java: - СПб.: «Питер», 2015.- 1168 с.

Допоміжна література

1. Wang A. J., Qian K. Component-oriented programming: - John Wiley & Sons. Inc., 2005.–334p.
2. Forman I.R., Forman N. Java Reflection in Action: - Manning Publications Co., 2005 – 297 p.
3. Harold E. R. Java I/O: - O'Reilly, 1999 – 596 p.
4. Гарольд Э.Р., Минс У.С. XML Справочник. - СПб: Символ-плюс, 2002 – 576 с.
5. Englander R. Developing JAVA Beans: - O'Reilly, 1997 – 231 p.
6. Harold E. R. Java Network Programming: - O'Reilly, 2005 – 735 p.
7. Grosso W. Java RMI: - O'Reilly, 2001 – 572 p.
8. Doss G. M. CORBA Networking with Java: - Wordware Publishing, 1999 – 327 p.

10. Посилання на інформаційні ресурси в Інтернеті, відео-лекції, інше методичне забезпечення

1. Java Platform Standard Edition 8 Documentation: [Електроний ресурс]// Oracle and/or its affiliates, 1993-2017. URL: <https://docs.oracle.com/javase/8/docs>.
2. Java API for XML Processing (JAXP): [Електроний ресурс]// Oracle and/or its affiliates, 1993-2017. URL: <https://docs.oracle.com/javase/tutorial/jaxp/>
3. Processing XML with Java: [Електроний ресурс]// Elliotte Rusty Harold, 2001-2003. URL: <http://www.cafeconleche.org/books/xmljava/>
4. XML Technology: [Електроний ресурс]// W3C, 2015. URL: <http://www.w3.org/standards/xml/>
5. DTD Schema: [Електроний ресурс]// W3C, 2015. URL: <http://www.w3schools.com/DTD/>
6. Namespaces in XML: [Електроний ресурс]// W3C, 2015.. URL: <http://www.w3.org/TR/xml-names/>
7. XML Schema: [Електроний ресурс]// W3C, 2015. URL: <http://www.w3.org/XML/Schema.html>
8. Document Object Model (DOM): [Електроний ресурс]// W3C, 2015. URL: <http://www.w3.org/DOM/>
9. Trail: JavaBeans(TM): [Електроний ресурс]// Oracle and/or its affiliates, 1993-2017. URL: <https://docs.oracle.com/javase/tutorial/javabeans/>
10. JavaBeans: [Електроний ресурс]// Oracle and/or its affiliates, 1993-2017. URL: <http://java.sun.com/products/javabeans>
11. Trail: Custom Networking: [Електроний ресурс]// Oracle and/or its affiliates, 1993-2017. URL: <https://docs.oracle.com/javase/tutorial/networking/>
12. Trail: RMI: [Електроний ресурс]// Oracle and/or its affiliates, 1993-2017. URL: <https://docs.oracle.com/javase/tutorial/rmi/>
13. Java Distributed Computing: [Електроний ресурс]// O'Reilly & Associates, 2001. URL: <http://docstore.mik.ua/orelly/java-ent/dist/>
14. Java Enterprise in Nutshell: [Електроний ресурс]// O'Reilly & Associates, 2001. URL: <http://docstore.mik.ua/orelly/java-ent/jenut/index.htm>

15. Enterprise Java Beans: [Электроний ресурс]// O'Reilly & Associates, 2001. URL: <http://docstore.mik.ua/oreilly/java-ent/ebeans/index.htm>
16. Построение распределенных систем на Java: [Электроний ресурс]// НОУ «ИНТУИТ», 2003 – 2017. URL: <http://www.intuit.ru/studies/courses/633/489/info>.