



JAVA

ASSOCIATED

**(Об'єктне проектування
на *JAVA*)**

**за вимогами корпоративного іспиту
*1Z0-808 "Oracle Certified Associated,
Java SE8 Programmer"***

Autumn Java Camp 2021

Java Associated

(об'єктне проєктування на Java)

Програма навчання

1. Ключове слово “*static*”. Локальні змінні, змінні об'єкту та класу.
2. Особливості передавання в метод в якості аргументів примітивів та посилань.
3. Типи методів. Конструктори. Перезавантажені конструктори.
4. Перезавантаження методів.
5. Метод *main()*. Аргументи в методі *main()*.
6. Ініціалізація змінних. Блоки ініціалізації. Порядок ініціалізації.
7. Пакети. *Import*. *Static import*. Клас *Math*.
8. Поняття наслідування. Відношення *has-a*, *is-a*. UML-діаграми. Класи *Calendar*, *GregorianCalendar*, *Date*.
9. Модифікатори доступу.
10. Ключове слово *this*.
11. Ключове слово *super*. Клас *Object*.
12. Перевизначення методів. Метод *toString()*.
13. Поліморфні посилання. Поліморфізм. Метод *equals()*.
14. *Covariant returns*.
15. Приведення типів. Оператор *instanceof*.
16. Ключове слово *final*.
17. Абстрактні класи.
18. Інтерфейси. Інтерфейси *Comparable*, *Comparator*.
19. Внутрішні класи (регулярні, локальні, анонімні, статичні).
20. Функціональні інтерфейси та лямбда-вирази.
21. *Exceptions*.
22. Робота з текстовими файлами.
23. *Strings*, *StringBuffers*, *StringBuilders*.
24. *Wrappers*, *autoboxing*, *autounboxing*.
25. *Enums*.
0. *JavaFX*.
26. Клас *Arrays*.
27. *Var-args*.
28. *Garbage collector*.
29. API Java SE8 для роботи із датами та часом. Класи пакету *java.time*: *LocalDate*, *LocalTime*, *LocalDateTime*, *Period*. Клас *java.time.format.DateTimeFormatter*.

1. Ключове слово “static”. Локальні змінні, змінні об’єкту та класу.

Відео: 01_Static.

Література: [Шилдт] – ст.202...204 (англ. – р.236...238),
[Эккель] – ст. 60, 61 (англ. – р.51, 52),
[Хорстманн] – ст. 151...153 (англ. – р.223...226),
[Вязовик] – ст. 241...245,

Тести обов’язкові (10): [Sanghera] – ch.3, # 5,
[Heller] – ch.3, # 20,
[Green] – # 34,
[Jaworsky] – p.72, # 24,
[Reese] – ch.2, # 1, 3,
ch.6, # 2,
[Mala Gupta] – MockExam, # 41, 86,
[Boyarsky] - ch.4, # 11.

Програмне завдання. Згідно вимог шаблону MVC створити класи для моделювання ремонту автомобілів двома способами: власноруч водієм та на станції техобслуговування (класи Car, Driver, AutoserviceStation),

2. Особливості передавання в метод в якості аргументів примітивів та посилань.

Відео: 02_Parameters.

Література: [Шилдт] – ст.193...195 (англ. – р.225...227),
[Хорстманн] – ст. 157...163 (англ. – р.231...238),

Тести обов’язкові (11): [Sanghera] - ch.4, # 11,
[Sierra7] - ch.3, # 6, 12,
[Heller] – ch.1, # 6, 7,
p.491, # 47,
[Raposa] – ch.1, # 9,
[Mala Gupta] – MockExam, # 42, 68,
[Boyarsky] - AssesmentTest, # 6,
ch.4, # 16.

Програмне завдання. Створити клас-контролер з двома методами:

- для обчислення чотирьох математичних функцій над дробовим та цілим числами (*div* - ділення першого операнду на другий, *log10* - десятковий логарифм суми операндів, *log* - натуральний логарифм суми операндів, *pow* - піднесення першого операнду у степінь другого операнду), де результат буде присвоюватись першому операнду (використати стандартні методи класу *Math*),
- для множення на дробову константу одного з трьох полів об’єкту класу *Car*, кожне з яких (*price*, *weight*, *speed*) представлено дробовим значенням.

Першим параметром в методах передавати відповідний рядок (*String*), що за допомогою оператора *switch/case* буде визначати необхідну для виконання функцію (в першому методі) або необхідне поле для множення (в другому методі). Таким чином, в кожний з методів необхідно передавати три параметри: для першого методу – 1)рядок потрібної функції, 2)перший операнд, 3)другий операнд; для другого методу – 1)рядок, що визначає потрібне поле, 2)об'єкт класу *Car*, 3)операнд-множник.

3. Типи методів. Конструктори. Перезавантажені конструктори.

Відео: 03_Constructors.

Література: [Шилдт] – ст.177...180, 188...193 (англ. – р.204...208, 219...225),
[Эккель] – ст.116...118, 125...126 (англ. – р.107...109, 114...115),
[Хорстманн] – ст. 165...167 (англ. – р.239...243),
[Мухортов] – с.50,
[Вязовик], с.195...197.

Тести обов'язкові (8): [Sanghera] – ch.3, # 1,
[Reese] – ch.6, # 4,
[Ganesh] – ch.3, # 1, 2,
[Mala Gupta] – MockExam, # 22, 29,
[Boyarsky] - ch.4, # 19,
[Sierra7] - ch.3, # 9.

4.Переавантаження методів.

Відео: 04_OverloadMethods.

Література: [Шилдт], с.185...188 (англ. – р.215...219),
[Эккель], с.118...125 (англ. – р.109...114),
[Хорстманн], с.164-165 (англ. – р.238-239),
[Вязовик], с.192...194, 210.

Тести обов'язкові (5): [Sierra7] - ch.2, # 7,
[Green], #52,
[Reese] – ch.6, # 3,
[Mala Gupta] – MockExam, # 24, 28.

5. Метод `main()`. Аргументи в методі `main()`.

Відео: 05_Main.

Література: [Шилдт], с.212...213 (англ. – р.249...250),

Тести обов'язкові (9): [Boyarsky] - ch.1, # 8, 9, 10,
[Green] – 29, 38,
[Reese] – ch.1, # 1,
[Mala Gupta] – MockExam, # 13, 70,
р. 28, Tale 1.3.

Програмне завдання. Із використанням циклічного оператора *for* та вкладеного оператора *switch/case* реалізувати введення командним рядком та відповідне виведення даних згідно таблиці:

WEEK_DAYS		MARKS		PLANETS	
Input	Print	Input	Print	Input	
1	Monday	0	Unsatisfactory	1	Mercury
2	Tuesday	1		2	Venus
3	Wednesday	2		3	Earth
4	Thursday	3	Satisfactory	4	Mars
5	Friday	4	Good	5	Jupiter
6	Saturday	5	Excellent	6	Saturn
7	Sunday			7	Uran
				8	Neptune

Наприклад, введеним даним:

MARKS 3 5 2 3 4 4 3 4

має відповідати виведення:

Satisfactory
Excellent
Unsatisfactory
Satisfactory
Good
Good
Satisfactory
Good

Реалізувати код в шаблоні MVC (для даного завдання клас View можна не створювати, а його функції доручити іншим класам). Використати масиви Strings із заданими в таблиці відповідними текстовими значеннями. Утворити відповідні ArrayList<Strings> weekDays, ArrayList<Strings> marks, ArrayList<Strings> planets.

6. Ініціалізація змінних. Блоки ініціалізації. Порядок ініціалізації.

Відео: 06_Initialization.

Література: [Эккель], с.137...146 (англ. – p.125...133),
 [ukrSierra6] – 10_Блоки ініціалізації.

Тести обов'язкові (8): [Heller] – ch.1, # 15,
 ch.3, # 5,
 [Boyarsky] - ch.1, # 21,
 ch.4, # 22, 23,

[Yaworsky], p.73, # 31, 51,
[Mala Gupta] – ch.5, # 4.

Тести додаткові (5): [Green], # 28,
[Sanghera] – p.333, # 1,
[Boyarsky] - ch.1, # 11, 12, 13.

7. Пакети. Import. Static import. Клас Math.

Відео: 07_PackagesImport.

Література: [Шилдт], с.249 ... 251, 257, 258, 406 ... 409 (англ. – p.298...300,
307...309, 494...497),
[Эккель], с.153...155 (англ. – p.146...148),
[Хорстманн], с.172...177 (англ. – p.249...257),
[Вязовик], с.148...151, 156...161.

Тести обов'язкові (12): [Boyarsky] - ch.1, # 5, 6, 7,
ch.4, # 10, 12,
[Sierra5], ch.10, #3, 4,
[Raposa], ch.2, #7,
AssesmentTest, #7,
[Ganesh] – MockExam1, # 10, 11,
MockExam2, # 30 (в тесті є неточності),

Тести додаткові (13): [Mala Gupta] – MockExam, # 9, 20, 27,
ch.1, # 8,
[Sanghera], p.335, #6,
p.337, #12.
[Heller], ch.1, #4,11,12,
p.476, #2,
[Boyarsky] - ch.1, # 14, 18,
ch.4, # 14.

8. Поняття наслідування. Відношення has-a, is-a. UML- діаграми. Класи Calendar, GregorianCalendar, Date.

Відео: 08_InheritanceDates.

Література: [Шилдт], с.221...223, 224...226, 738...743 (англ. – p.260...263,
265...267, 893...899),
[Хорстманн], с.129...130 (англ. – p.193...196),
[Мухортов], с.70...71.

Тести обов'язкові (10): [Sanghera], p.354, #60,
[Raposa], ch.1, #10,

[Mala Gupta] – MockExam, # 33,
[Jaworsky], p.112, #17,
[Heller], ch.6, #3, 4,
[Sierra7] – ch.1, # 9,
[SierraExam]: PE1, #28,
PE2, # 43,
PE3, #11,

Тести додаткові (8): [Sanghera], ch.5, #12, 13, 14,
[Jaworsky], p.30, #15,
[Reese] – ch.7, # 1,
[Ganesh] – ch.3, # 3,
ch.5, # 4, 5.

Програмні завдання.

1. Вивести на екран дні тижня (Sunday, Monday, etc.) Вашого дня народження в наступні 5 років.
2. Визначити кількість днів тривалості Другої світової війни (враховувати дні початку та завершення війни).
3. Для ієрархії класів:

Транспортний засіб => Літак, Автомобіль, Корабель;

Літак => Пасажирський літак, Транспортний літак, Винищувач,

Автомобіль => Легковий автомобіль, Автобус, Вантажний автомобіль,

Корабель => Пасажирський лайнер, Буксир, Танкер –

оголосити в **кожному** класі 1-3 найбільш доцільних поля

9. Модифікатори доступу.

Відео: 09_Modifiers.

Література: [Шилдт], с.252...257, 199...203 (англ. – p.302...307, 231...236),
[Эккель], с.159...163, 164...168 (англ. – p.145, 153...158),
[Sierra6], p.34...36,
[Жданов], с.7, 8,
[Вязовик], с.174...184,

Тести обов'язкові (10): [SierraExam], PE2, #18,
PE4, #16,
[Sierra7] – ch.1, # 5,
[Boyarsky] - ch.4, # 7,
[Raposa], ch.1, #8,
[Heller], ch.8, #4,
[Sanghera], ch.4, #8, 9, 10,
[Green], #25,

Тести додаткові (19): [Heller], ch.3, #6,
AssesmentTest, # 5,
[Raposa], ch.2, #4, 6,
ch.6, #8, 14,
[Jaworsky], p.72, #23,
[Reese] – ch.6, # 5, 6,
[Sanghera], ch.5, #6,
[Mala Gupta] – MockExam, # 10, 11,
ch.1, # 9, 10, 11,
ch.3, # 1, 2, 10,
[Boyarsky] - ch.4, # 6.

Програмне завдання.

В ієрархії класів: Транспортний засіб => ... в кожному класі ввести методи getters/setters для всіх полів класу. Полям всіх класів надати доступ рівня private, методам getters/setters – доступ рівня public для всіх класів, крім класу Fighter, в якому змінювати значення полів можуть тільки класи одного пакету, а читати значення полів – також і підкласи класу Fighter з інших пакетів.

10. Ключове слово this.

Відео: 10_This,

Література: [Шилдт], с.180...181, 409...411, (англ. – p.208...209, 498...500),
[Эккель], с.126...129, (англ. – p.116...119),
[Вязовик], с.246...248,

Тести обов'язкові (10): [Heller], ch.3, #17,
[SierraExam], PE2, #2,
[Ganesh] – MockExam2, # 3,
[Mala Gupta] – MockExam, # 34, 78,
p.144, Tale 3.2,
ch.3, # 9,
[Boyarsky] - AssesmentTest, # 12,
ch.4, # 18, 20.

11.Ключове слово super. Клас Object.

Відео: 11_Super,

Література: [Шилдт], с.228...236, 248...249, (англ. – p.268...281, 295...296),
[Эккель], с. 195...197, 208...210, (англ. – p.189...190, 204...206),
[Хорстманн], с.217, (англ. – p.314),
[Жданов], с.24...26,

Тести обов'язкові (15): [Sanghera], ch.3, #3, 6, 8,
p.336, #9, 10,
p.341, #23,
p.342, #24,
[SierraExam], AsT1, #1,
AsT2, #7,
PE1, #47,
PE2, #31, 47,
PE3, #23,
PE4, #19, 60,

Тести додаткові (17): [Raposa], ch.2, #20, 25,
[Boyarsky], ch.4, # 21,
ch.5, # 2,
[Ganesh], MockExam1, # 6, 8, 9,
[Jaworsky], p.111, #1,
[Heller], ch.6, # 6, 7,
p.487, #37,
p.488, #38,
[Mala Gupta] – MockExam, # 7, 75, 77,
p.303, Tale 6.1,
p.323, Tale 6.3.

Програмне завдання.

В ієрархії класів: Транспортний засіб => ... в кожному класі ввести конструктор для ініціалізації всіх полів класу.

12.Перевизначення методів. Метод *toString()*.

Відео: 12_Overriding,

Література: [Шилдт], с.236...239, (англ. – p.282...285),
[Вязовик], с.248, 250,
[Жданов], с.21...22,

Теми, відсутні у відео – перевизначення *private* методів: [Эккель], с.207...208,
перевизначення *static* методів: [Жданов], с.21,

Тести обов'язкові (9): [SierraExam], PE3, #57,
[Sanghera], ch.4, #5, 6,
ch.5, #15,
p.342, #25,
[Jaworsky], p.71, #1,

[Raposa], ch.2, #15, 23,
ch.6, #7,

Тести додаткові (8): [Heller], ch.3, #7, 9, 10, 11, 16,
ch.4, #5,
ch.6, #11,
p.487, #36.

Програмне завдання.

Для ієрархії класів Транспортний засіб =>... ввести до кожного класу метод *toString*.

13.Поліморфні посилання. Поліморфізм. Метод *equals()*.

Відео: 13_Polymorphism,
13a_Equals

Література: [Шилдт], с.227...228, 240...243, (англ. – р. 267...268, 285...290),
[Эккель], с. 201...208, 214...216, 217...218, (англ. – р. 196...204,
210...211, 212...213),
[Вязовик], с.259...267,
[Хорстманн], с.197...207, 218...219, (англ. – р.284...300, 315...317),

Тести обов'язкові (9): [SierraExam], PE2, #20,
PE4, #25,
[Sanghera], p.337, #13,
p.338, #15,
[Heller], p.490, #45,
[Raposa], AsT, #33,
ch.6, #6,
[Mala Gupta] – MockExam, # 6, 71.

Програмне завдання.

1. Для ієрархії класів Транспортний засіб => ... ввести до кожного класу метод *equals*.
2. В *ArrayList*, що містить не менше 3 об'єктів класу *Car*, що в свою чергу, містить поля *String model*, *String owner*, *int price*, *int produceYear*, визначити факт наявності в колекції автомобілів власника "Serg". Завдання визначити двома способами:
– методом *contains* класу *ArrayList*;
– іншим нестандартним методом, створеним розробником в класі *Car*.
3. Для завдання формування крапкових геометричних фігур (заняття 6 курсу "Вступ до Java") розробити ефективну ієрархію класів *Model*.

14.Covariant returns.

Відео: 14_Covariant,

Література: [Эккель], с.216...217, (англ. – р. 211...212),
[Жданов], с.39,

Тести обов'язкові (6): [SierraExam], PE2, #4,
[Sanghera], p.341, #22,
[Heller], p.478, #7,
[Sierra5], ch.2, #8, 10.

15. Приведення типів. Оператор instanceof.

Відео: 15_Casting,

Література: [Шилдт] – ст.399...402, (англ. – р. 486...489),
[Хорстманн], с.209 - 211, (англ. – р. 301...305),
[Вязовик] – с.222...226,
[Жданов] – с.12, 13,

Тести обов'язкові (19): [SierraExam], PE1, # 33, 51,
PE2, #17, 35, 54,
PE3, #15, 16, 55,
PE4, #8, 51,
[Sierra5], ch.2, #13, 14,
ch.3, #7,
[Sierra6], ch.2, #8, 9,
[Raposa], ch.6, #15, 16,
[Heller], p.486-487, #34, 35,
Тести додаткові (5): [Heller], ch.2, # 17,
ch.4, # 1, 6, 20,
[Jaworsky], p.56, #6.

16. Ключове слово final.

Відео: 16_Final,

Література: [Шилдт] – ст.204...205, 246...247, (англ. – р. 239, 293...295),
[Хорстманн], с.207...208, (англ. – р. 300...301),
[Эккель], с.188 - 195, (англ. – р. 182...189),

Тести обов'язкові (10): [SierraExam], PE1, #4, 40,
PE2, #33,
PE4, #32, 37,
[Sierra5], ch.2, #2, 6,
[Raposa], ch.3, #5,
[Sanghera], p.338, #14,
p.339, #18,

Тести додаткові (9): [Raposa], ch.6, #12,
[Heller], AsT, #7, 9, 10,
ch.3, # 3, 14, 18, 19,

[Green], #6.

Програмне завдання.

За допомогою ключового слова `final`, а також методу `unmodifiableList` класу `Collections` зробити неможливим будь-яку зміну даних в студентській групі після завершення навчання. Задіяти клас `Student` з полями `String name`, `ArrayList<Integer> marks`, клас `Group` з полем `ArrayList<Student> students`.

17. Абстрактні класи.

Відео: 17_Abstract,

Література: [Шилдт] – с.242...245, 247, (англ. – р. 290...293, 294...295),
[Эккель] – с.221 - 224, (англ. – р. 219...222),
[Хорстманн] – с.211...216, (англ. – р. 305...312),
[Вязовик] – с.250...253,
[Жданов] – с.8...9,

Тести обов'язкові (8): [SierraExam], PracticeExam 2, # 15,
PracticeExam 4, # 1, 23,
[Sierra5], ch1, # 2,
[Heller], ch.4, #8, 9, 10,
[Green], # 4,

Тести додаткові (9): [Raposa], ch.6, #9, 10, 11, 19, 20,
AsT, # 17,
[Jaworsky], p.15, #12,
p.72, #21,
[Heller], ch.3, # 12.

18. Інтерфейси. Інтерфейси Comparable, Comparator.

Відео: 18_Interfaces,
18a_Compare

Література: [Шилдт] – с.259...263, 268...271, 628, 686,
(англ. – р. 310...315, 321...325, 762, 830...831),
[Хорстманн] – с. 274...275, 280...282, 289...290,
(англ. – р. 395...397, 404...407, 415...417),
[Эккель] – с.224...227, 232...235, 238...239,
(англ. – р. 222...225, 230...233, 235...236),
[Вязовик] – с.253...259,
[Жданов] – с.10...11,

Тести обов'язкові (19): [SierraExam] – PracticeExam1, # 6, 17, 54,
PracticeExam2, # 12,
PracticeExam3, # 12, 22,
PracticeExam4, # 14, 27,

[Sierra5] – ch1, # 1, 3,
 ch2, # 3,
[Sierra6] – ch1, # 6,
 ch.4, #10,
[Sanghera], ch.5, #9, 10,
 practiceExam, # 7,
[Raposa], ch.2, #10, 24,
 AsT, # 24,

Тести додаткові (5): [Heller], ch.3, # 13,
 ch.4, # 12,
 AsT, # 15,
[Sanghera], ch.3, #4, 9.

Програмне завдання.

В проєкті з ієрархією класів Транспортний засіб => ... проаналізувати доцільність використання іншої ієрархії: Транспортний засіб => Пасажирський транспортний засіб, Вантажний транспортний засіб, а також заміни класів інтерфейсами з відповідними методами.

19.Внутрішні класи (регулярні, локальні, анонімні, статичні).

Відео: 19_Inner,

Література: [Шилдт] – с.207...210, (англ. – р. 242...245),
[Хорстманн] – с. 312...329, (англ. – р. 450...471),
[Эккель] – с.245...249, 251...256, 258...260,
 (англ. – р. 243...247, 249...254, 256...257),
[ukrSierra6] – 1_InnerClasses,
[Жданов] – с.31...33,

Тести обов'язкові (26): [SierraExam] – AsTest1, # 2,
 AsTest2, # 9,
 PracticeExam2, # 10, 36,
 PracticeExam3, # 30,
 PracticeExam4, # 47,

[Sierra5] – ch.8 – 1...12,
[Sierra6] – ch.8 – 7...9, 11, 12,
[Raposa] – ch.2 – 12, 22,

Тести додаткові (19): [Green] – 23, 44, 54,
 [Heller] – ch.6 – 8,9,15,17,
 [Jaworsky] – p.15 – 10,
 p.112 – 18, 34, 35,
 [Ganesh] – PreTest, #11,
 Mock1, #14, 15, 24,
 ch.4, #4, 5, 6,

[Sanghera] – ch.3 – 2.

Програмне завдання.

ArrayList з 6-8 об'єктів класу *Person* з полями *String name*, *String region*, *int birthYear*, *boolean isMale*, НЕ ВИКОРИСТОВУЮЧИ *default* та *static* методів інтерфейсу *Comparator*, посортувати в залежності від введеного *String* наступним чином:

- при введенні "1" – по *region*, при однакових *region*: *name* => *isMale* => *birthYear*,
- при введенні "2" – по *birthYear*, при однакових *birthYear*: *isMale* => *name* => *region*,
- при введенні "3" – по *isMale*, при однакових *isMale*: *name* => *region* => *birthYear*,
- при введенні будь-якого іншого *String* або відсутності введених *Strings* – по *name*, при однакових *name* => *region* => *birthYear* => *isMale* (цей спосіб сортування вважати основним із застосуванням *Comparable*).

20. Функціональні інтерфейси та лямбда-вирази

Відео: 20_Lambdas,

Література:

- статичні методи в інтерфейсах: [Шилдт] –ст.276...277,
- default-методи в інтерфейсах: [Шилдт] –ст.272...276,
[Хорстманн] – ст. 296...309,
[Уорбэртон] –ст. 63...69,
- поняття функціонального інтерфейсу та прості лямбда-вирази:
[Шилдт] –ст.459...490,
[Хорстманн] – ст. 288...292, 311...312,
[Уорбэртон] –ст. 20...25,
- стандартні функціональні інтерфейси пакету *java.util.function*:
[Шилдт] –ст.490...491, 789...792,
[Хорстманн] – ст. 309...311,
[Уорбэртон] –ст. 26.

Тести обов'язкові (5): [Boyarsky] – ch.4: 1, 26...29.

Програмні завдання.

1.Із використанням стандартних функціональних інтерфейсів з пакету *java.util.function* та лямбда-виразів реалізувати опрацювання масивів цілих чисел із утворенням колекцій:

- масив розділити на дві колекції:
 - від'ємних та додатніх чисел,
 - парних та непарних чисел,
 - визначити середні арифметичні двох масивів, утворити колекцію з елементів обидвох масивів, що знаходяться в межах між значеннями середніх арифметичних.
- 2.Із використанням стандартних функціональних інтерфейсів з пакету *java.util.function* та лямбда-виразів розділити кілька вхідних колекцій об'єктів класу *Car* із полем *int maxSpeed* на дві колекції з об'єктів, швидкість яких менше (перша колекція) та не менше (друга колекція) заданого значення.

21.Exceptions.

Відео: 21_Exceptions,

Література: [Шилдт], с.280...297,
[Хорстманн], с.339...355, 361...364,
[Эккель], с.310...316, 319...325, 328...351,
[ukrSierra6] – 2_Exceptions,
[Жданов], с.18...19,
[Вязовик], с.317...334.

Теми, відсутні у відео – try з ресурсами, catch із функцією OR:

[Шилдт] –ст.299...301,
[Хорстманн] – ст. 350...351, 355...366,

Тести обов'язкові (14): [SierraExam] – PracticeExam3, # 10, 17, 32, 34, 36, 37,
PracticeExam4, # 22, 31, 42, 53,
[Sierra7] – ch.7: 4...6, 12,

Тести додаткові (19): [Sierra5] – ch.5: 3, 6, 14,
[Sierra6] – ch.5: 7, 11, 13, 15, 16,
ch.8: 10,
[SierraExam] – AssTest1, #7,
AssTest2, #10,
PracticeExam1, # 9, 13, 19, 27, 36, 37,
PracticeExam2, # 13, 29.

22. Робота з текстовими файлами.

Література обов'язкова: [ukrSierra6] – 3_InputOutput.

Література додаткова:

[Шилдт], с.795...813, 826...833, 836...837, 839...840,
[Эккель], с.483...484, 487...491, 494...495, 497...499, 500...509,
[Хорстманн, т.2], с.71...79, 82...85, 92...94,
[Вязовик], с.505...511, 521...523, 525...527, 542...545, 546...549.

Тести обов'язкові (8): [Sierra5] - ch.6 - 3,12,
[Sierra6] - ch.6 - 10,
[SierraExam] - PracticeExam2 – 8,
PracticeExam3 – 41,
PracticeExam4 – 18, 44, 49.

Програмне завдання. Прочитати та обробити дані з трьох існуючих файлів та занести оброблені дані у файли, створені засобами Java.

1. Існуючі файли містять інформацію щодо довільних пляшок у наступному форматі:

N	Bottle	Volume	Material
1	Wine	0.75	Glass
2	Juice	0.25	Metal

...

2.Зміст оброблення даних з файлів:

- прочитати дані з існуючих файлів,
- утворити три файли, до яких занести пляшки місткістю: не більше 0.5 (перший файл), в межах від 0.51 до 0.99 (другий файл), не меншою за 1.0 (третій файл),
- пляшки у всіх утворених файлах посортувати згідно одного з трьох варіантів:
 - по Volume, при рівності – по Material, при рівності – по Bottle (основний спосіб),
 - по Bottle, при рівності – по Volume, при рівності – по Material,
 - по Material, при рівності – по Volume, при рівності – по Bottle.
- спосіб сортування задавати довільним способом (Scanner, масив String[] args методу main, тощо).

3.Передбачити виведення потрібної інформації у випадку виникнення Exceptions при роботі з файлами.

4.Для перетворення текстового рядка у масив Strings використати метод split() класу String. Для перетворення текстових змінних у числові використати метод parseDouble() класу Double.

23. Strings, StringBuffers, StringBuilders.

Відео: 22_Strings.

Література: [Шилдт], с.529...558,
[ukrSierra6] – 4_Strings.

Тести обов'язкові (12): [Sierra6] – ch.6 – 9,
[Sierra7] – ch.6 – 14,
[Boyarsky] – AssTest – 2,
ch.3 – 1, 2, 4...6, 8...11.

24. Wrappers, autoboxing, autounboxing

Відео: 23_Wrappers.

Література: [Шилдт], с.560...584,
[ukrSierra6] – 5_Wrappers.

Тести обов'язкові (11):[SierraExam] – PracticeExam3 – 45,
PracticeExam4 – 2, 5,
[Sierra5] - ch.4 – 3,
ch.5 – 13,
[Ganesh] – MockExam2 – 78,
PreTest – 3,
[Boyarsky] - ch.4, - 15,
[Raposa] - AsT, - 18,
ch.4 – 1,
[Sanghera] – PracticeExam – 42.

25. Enums

Відео: 24_Enums.

Література: [Шилдт], с.337...345, 626...627,
[ukrSierra6] – 6_Enums.

Тести обов'язкові (7): [SierraExam] – AssTest2 – 4,
PracticeExam1 – 3,
PracticeExam2 – 34,
PracticeExam3 – 7, 51,
PracticeExam4 – 9, 17,

Програмне завдання. В класі Fest (фестиваль) є два поля типу String: “Назва” та “Місто проведення”, а також третє поле “Місяць проведення” типу enum YearMonth. Забезпечити основне сортування об'єктів класу Fest по назвах, додаткові – по містах проведення та по місяцях проведення.

0. JavaFX

Відео: 0_JavaFX

Література: [Шилдт] – с.1277...1294.

Програмне завдання. На основі оболонки Java FX спроектувати:

- простий секундомір аналогічно завдання [Шилдт-Swing] - с.51,
- просту шифрувальну машину аналогічно завдання [Шилдт-Swing] - с.65.

26. Клас Arrays.

Відео: 25_ArraysClass.

Література: [ukrSierra6] – 7_ArraysClass.

Тести обов'язкові (7): [SierraExam] – AsTest2, # 1,
PracticeExam1, # 22, 39,
PracticeExam2, # 26, 40,
[Ganesh] – PreTest, #4, 17.

Тести додаткові (9): [MalaGupta] – ch.4, # 2, 3,
MockExam, # 52,
[Reese] – ch.4, #1, 2,
ch.6, # 1,
[Boyarsky] – ch.3, # 15, 16, 22.

27. Var-args

Відео: 26_VarArgs.

Література: [Шилдт] – с.215...219,
[ukrSierra6] – 8_Var-args.

Тести обов'язкові (5): [Sierra5], ch.3 – 3, 5,
[Raposa], ch.2 – 8,11,13.

Тести додаткові (2): [Heller], PracticeExam – 6,
[Raposa], AsmTest – 6.

28.Збирач сміття Garbage collector

Відео: 27_GarbageCollector.

Література: [Шилдт], с.181, 586...587,
[Эккель], ст.130...137,
[ukrSierra6] - 9_ GarbageCollector.

Тести обов'язкові (6): [Sierra5], ch.3 – 2, 10,
[Raposa], ch.1 – 7, 11, 15, 16.

Тести додаткові (7): [Heller], ch.1 – 8,19,
[Jaworsky], p.120 – 30...34.

Програмне завдання. Дано клас Car з полем String model (наприклад, “BMW”, “Opel”, “Ford”, etc). Використовуючи перевизначення методу finalize(), забезпечити виведення на екран значень полів model трьох об'єктів класу Car після скидання в null їх посилань:

```
Car car1 = new Car("BMW");  
// ...  
car1 = null;
```

29. API Java SE8 для робот із датами та часом. Класи пакету java.time: LocalDate, LocalTime, LocalDateTime, Period. Клас java.time.format.DateTimeFormatter.

Література: [Шилдт], с.1177...1182,
[Хорстманн, т.2], с. 353 – 376 (гл.6).

Тести обов'язкові (17): [Boyarsky] – ch.3: 28-33,
[Mughal] – ch.11: 2– 8, 10-13.

Програмне завдання. Додати до коду, наведеного в [Хорстманн, т.1, с.136...138], наступні ДОДАТКОВІ можливості (завдання режимів – за вибором студента):

- завдання довільних року та місяця, крім поточного місяця,
- вибір формату виведення: Sun-Mon-Tue-Wen-Thu-Fri-Sat або
Пн-Вт-Ср-Чт-Пт-Сб-Нд.

ЛІТЕРАТУРА

1. Шилдт Герберт. Java. Полное руководство. 10-е изд.: Пер. с англ. – СПб.: ООО “Альфа-книга”, 2018. -1488 с.
2. Хорстманн Кей С. Java. Библиотека профессионала. 11-е изд.: – СПб.: Диалектика, 2019. Т.1. Основы. - 864 с.
3. Хорстманн Кей С. Java. Библиотека профессионала. 11-е изд.: – СПб.: Диалектика, 2020. Т.2. Расширенные средства программирования. - 864 с.
4. Эккель Брюс. Философия Java. – СПб.: Питер, 2009. -640 с.
5. Вязовик Н.А. Программирование на Java. - М.: ИНТУИТ, 2016.
6. Іщеряков С.М. Комп’ютерна схемотехніка.–Івано-Франківськ: Полум’я,2011.– 114 с.
7. Уорбэртон Р. Лямбда-выражения в Java 8. Функциональное программирование – в массы. – М.: ДМК Пресс, 2014. – 195 с.
8. Шилдт Герберт. SWING: руководство для начинающих. – М.: ООО “И.Д.Вильямс”, 2007. -704 с.
9. S.G.Ganesh, Tushar Sharma. Java SE7 Programmer Exams 1Z0-804 and 1Z0-805. A Comprehensive OCPJP 7 Certification Guide.– APRESS,New York,2012.-644 p.
- 10.Philip Heller, Simon Roberts. Complete Java 2 Certification Study Guide. 5-th edition. – SYBEX, Inc., Alameda, 2005.- 560 p.
- 11.Raposa, Richard F. SCJP Sun Certified Programmer for Java Platform, SE6, Study Guide. – Wiley Publishing, Inc., Indianapolis, 2009.- 583 p.
- 12.Khalid A.Mughal, Rolf W.Rasmussen. A Programmer’s Guide to Java SE8 Oracle Certified Associated (OCA). – Addison-Wesley Publishing, 2017.
- 13.Kathy Sierra, Bert Bates. SCJP Sun Certified Programmer for Java 5 Study Guide. – McGraw Hill Osborne, New York, 2006. -924 p.
- 14.Kathy Sierra, Bert Bates. SCJP Sun Certified Programmer for Java 6 Study Guide. – McGraw Hill Osborne, New York, 2008. -852 p.
- 15.Kathy Sierra, Bert Bates. OCA/OCJP Java SE7 Programmer I & II Study Guide (Exams 1Z0-803 & 1Z0-804) (Oracle Press).– McGraw Hill Osborne, New York, 2015. -1088 p.
- 16.Bert Bates, Kathy Sierra. OCP Java SE6 Programmer Practice Exam.– McGraw Hill Osborne, New York, 2011. -422 p.
- 17.Jamie Jaworsky. Java 2 Certification Training Guide. – New Riders Publishing, Indianapolis, 1999.- 399 p.
- 18.Mala Gupta. OCA Java SE7 Programmer I Certification Guide. Prepare for the 1Z0-803Exam. – Manning, Shelter Island, 2013.- 528 p.
- 19.Paul Sanghera. SCJP Exam for J2SE5. A Concise and Comprehensive Study Guide for the Sun Certified Java Programmer Exam. – APRESS, New York,2006.- 423 p.
- 20.Jeanne Boyarsky, Scott Selikoff. OCP Oracle Certified Professional Java SE8 Programmer II. Study Guide. Exam 1Z0-809- John Wiley & Sons, Inc., Indianapolis, 2016.- 669 p.
- 21.Richard M.Reese. Oracle Certified Associate Java SE7 Programmer Study Guide. – Packt Publishing, Birmingham, 2012.- 313 p.

Програму розроблено ТОВ “Інновейшн Девелопмент Фаундейшн”