BONUS #1

>_ PLAN NAUKI JAVY

12 tygodni: od zera do pierwszych projektów



Tematy, mapy myśli i zadania na każdy tydzień

PRAWA AUTORSKIE

© 2026 Bonus #1 · Plan nauki Javy. Wszelkie prawa zastrzeżone.

Ten materiał jest przeznaczony wyłącznie do osobistego użytku edukacyjnego osoby, która go nabyła. Kopiowanie, udostępnianie i rozpowszechnianie całości lub fragmentów jest dozwolone tylko w zakresie wynikającym z zakupionej licencji.

Plan ma charakter pomocniczy i wspiera samodzielną naukę programowania. Efekty zależą od regularnej nauki i praktyki każdego ucznia.

Java jest znakiem towarowym należącym do jego właściciela. Ten materiał jest niezależny i nie jest oficjalnie powiązany z właścicielem znaku.

>_ SPIS TREŚCI

Na start

Twoja droga do pierwszych projektów — 03

Jak korzystać z planu — 04

Mapa 12 tygodni — 05

Zanim zaczniesz — 06

Etap 1

Etap 1: Podstawy — 07

Tydzień 01: Pierwszy program w Javie — 08

Tydzień 02: Wejście, wyjście, zmienne i typy — 14

Tydzień 03: Operatory i konwersje — 20

Etap 2

Etap 2: Logika — 26

Tydzień 04: Decyzje w kodzie — 27

Tydzień 05: Pętle i powtórzenia — 33

Tydzień 06: Metody i porządek w kodzie — 39

Etap 3

Etap 3: Dane i obiekty — 45

Tydzień 07: Tablice i Stringi — 46

Tydzień 08: Kolekcje i przetwarzanie danych — 52

Tydzień 09: Klasy i obiekty — 58

Etap 4

Etap 4: Tematy zaawansowane i projekty — 64

Tydzień 10: Dziedziczenie, polimorfizm i interfejsy — 65

Tydzień 11: Wyjątki, pliki, generyki, lambdy i Streamy — 71

Tydzień 12: Powtórka i pierwszy własny projekt — 77

Na koniec

Tablica postępów: 12 tygodni — 83

Ocena końcowa — 84

Karta Twojego projektu — 85

Co dalej po 12 tygodniach — 86

Gratulacje! — 87

TWOJA DROGA DO PIERWSZYCH PROJEKTÓW

Najtrudniejsze w nauce programowania nie jest samo programowanie. Najtrudniejsze jest pytanie: „Od czego zacząć i co dalej?“. Ten plan odpowiada na nie za Ciebie.



Przez 12 tygodni przejdziesz przez cztery etapy: Podstawy, Logikę, Dane i obiekty oraz Zaawansowane tematy z projektami. Każdy tydzień ma jasne cele, listę tematów, mapy myśli do przerobienia, plan na pięć dni i osiem krótkich zadań.

Nie musisz niczego wymyślać. Otwierasz plan, sprawdzasz, który to dzień, i robisz jedną konkretną rzecz. Pod koniec tygodnia zaznaczasz, co już potrafisz, i widzisz swój postęp czarno na białym.

Co zyskasz po 12 tygodniach

Napiszesz własne programy w Javie, zrozumiesz programowanie obiektowe i ukończysz swój pierwszy samodzielny projekt.

JAK KORZYSTAĆ Z PLANU

1

Zacznij od przeglądu tygodnia

Sprawdź cele, tematy i mapy myśli, które przerobisz w tym tygodniu.

2

Ucz się według planu dnia

Jedna sesja, jeden temat. Bez skakania po materiałach.

3

Poznaj pułapki

Dowiesz się, jakie błędy popełniają początkujący i jak ich uniknąć.

4

Zrób osiem zadań

Pisz kod samodzielnie i zaznaczaj każde ukończone zadanie.

5

Sprawdź postępy

W piątek lub w weekend wypełnij kontrolę postępów i dziennik nauki.

Twój rytm nauki



5 sesji w tygodniu

45–60 minut na sesję

weekend na powtórkę **lub odpoczynek**

masz mniej czasu? **Rozłóż tydzień na dwa**

MAPA 12 TYGODNI

Etap 1: Podstawy

01 Pierwszy program w Javie

02 Wejście, wyjście, zmienne i typy

03 Operatory i konwersje

Etap 2: Logika

04 Decyzje w kodzie

05 Pętle i powtórzenia

06 Metody i porządek w kodzie

Etap 3: Dane i obiekty

07 Tablice i Stringi

08 Kolekcje i przetwarzanie danych

09 Klasy i obiekty

Etap 4: Tematy zaawansowane i projekty

10 Dziedziczenie, polimorfizm i interfejsy

11 Wyjątki, pliki, generyki, lambdy i Streamy

12 Powtórka i pierwszy własny projekt

Każdy tydzień buduje na poprzednim. Nie przeskakuj etapów.

ZANIM ZACZNIESZ

Dobre przygotowanie zajmie Ci jeden wieczór, a oszczędzi wielu frustracji w pierwszym tygodniu.

Lista przygotowań

- [ ] Mam komputer z dostępem do internetu i uprawnieniami do instalowania programów.
- [ ] Wybrałem lub wybrałam środowisko programistyczne (IDE).
- [ ] Utworzyłem lub utworzyłam folder „nauka-javy” na wszystkie zadania.
- [ ] Mam pod ręką 150 map myśli i Bonus #3 · Szybki przewodnik po składni Javy.
- [ ] Wpisałem lub wpisałam sesje nauki do kalendarza.
- [ ] Przygotowałem lub przygotowałam zeszyt na notatki i pytania.

Moje cele

Za 12 tygodni chcę umieć:

_____

_____

Mój harmonogram

Poniedziałek, godz.: _____

Wtorek, godz.: _____

Środa, godz.: _____

Czwartek, godz.: _____

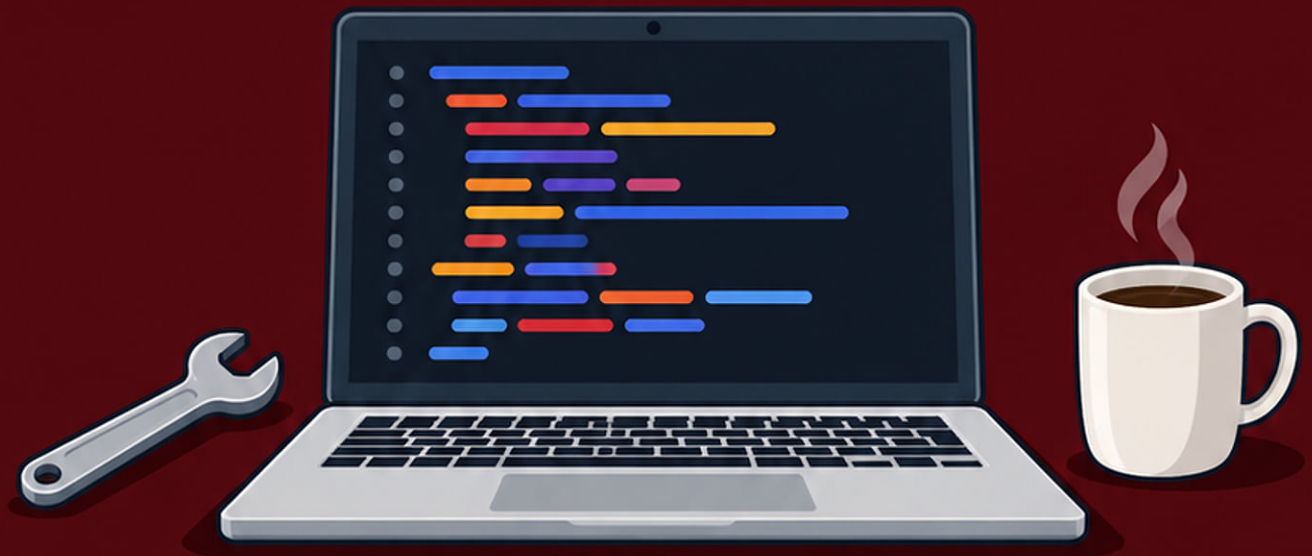
Piątek, godz.: _____

1

ETAP 1

PODSTAWY

Tygodnie 1–3: instalacja, pierwszy program, zmienne, typy i operatory



TYDZIEŃ 01

01 PIERWSZY PROGRAM W JAVIE

Cele tygodnia

Zainstalujesz narzędzia, uruchomisz pierwszy program i zrozumiesz, z czego składa się każda aplikacja w Javie.

Tematy

- Czym są JDK i JVM
- Środowisko programistyczne (IDE)
- Budowa programu: klasa i metoda main
- Wypisywanie tekstu: print i println
- Komentarze jedno- i wielowierszowe
- Zasady składni: średniki, nawiasy klamrowe, wielkość liter

Mapy do przerobienia

- mapy o instalacji i środowisku pracy
- mapy o składni i budowie programu
- mapy o komentarzach
- mapy o wypisywaniu danych na ekran

5 sesji po 45–60 minut

Wskazówka

Trzymaj pod ręką **Bonus #3 · Szybki przewodnik po składni Javy**. Zajrzyj do niego, gdy zapomnisz, jak coś zapisać.



TYDZIEŃ 01

PLAN DZIEŃ PO DNIU

Dzień 1	**Instalacja narzędzi** Zainstaluj JDK i IDE. W terminalu wpisz java -version i sprawdź, czy widzisz numer wersji.	45 min
Dzień 2	**Pierwszy program** Przerób mapy o budowie programu. Utwórz projekt, napisz program powitalny i uruchom go.	60 min
Dzień 3	**Wypisywanie tekstu** Przerób mapy o wypisywaniu danych. Wykonaj zadania 01–03.	60 min
Dzień 4	**Komentarze i składnia** Przerób mapy o komentarzach i składni. Wykonaj zadania 04–06.	60 min
Dzień 5	**Utrwalenie** Wykonaj zadania 07–08, a potem wypełnij kontrolę postępów.	60 min

Weekend: przejrzyj mapy z tego tygodnia albo po prostu odpocznij.

TYDZIEŃ 01

PUŁAPKI I WSKAZÓWKI

```
public class Main {
    public static void main(String[] args) {
        // mój pierwszy program
        System.out.println("Cześć, Java!");
    }
}
```

Zła nazwa pliku

Klasa publiczna Main musi być zapisana w pliku Main.java. Inna nazwa pliku oznacza błąd kompilacji.

Brak średnika

Każda instrukcja kończy się średnikiem. Gdy go zabraknie, kompilator wskaże wiersz z błędem.

Wielkość liter

Java rozróżnia wielkie i małe litery. Zapis system.out.println nie zadziała.

Wskazówka

Czytaj komunikat błędu od pierwszego wiersza. Zwykle podaje numer wiersza, w którym trzeba szukać problemu.

TYDZIEŃ 01

ZADANIA TYGODNIA (1/2)

Pisz kod samodzielnie. Zaglądaj do map tylko wtedy, gdy utkniesz.

ZADANIE 01 — Pierwsze uruchomienie

Utwórz nowy projekt, napisz program, który wypisze „Cześć, Java!”, i uruchom go w IDE.

ZADANIE 02 — Wizytówka

Wypisz w trzech osobnych wierszach swoje imię, swoje miasto i powód, dla którego uczysz się Javy.

ZADANIE 03 — print kontra println

Wypisz jedno zdanie za pomocą trzech instrukcji print, a potem trzech instrukcji println. Porównaj wyniki.

ZADANIE 04 — Prostokąt z gwiazdek

Wypisz na ekranie prostokąt z gwiazdek o szerokości 5 znaków i wysokości 3 wierszy.

Każde zadanie zapisz w osobnym pliku w folderze „nauka-javy”.

TYDZIEŃ 01

ZADANIA TYGODNIA (2/2)

05

ZADANIE 05 — Komentarze w praktyce

Dodaj do programu z zadania 02 komentarz jednowierszowy i komentarz wielowierszowy, które opisują, co robi kod.

06

ZADANIE 06 — Znaki specjalne

Użyj znaków \n i \t, aby wypisać prosty plan tygodnia: nazwę dnia i godzinę nauki w każdym wierszu.

07

ZADANIE 07 — Polowanie na błędy

Celowo usuń średnik i nawias klamrowy oraz zmień wielkość jednej litery. Zapisz, jaki komunikat pokazał kompilator przy każdym błędzie.

08

ZADANIE 08 — Rysunek ze znaków ⭐

Narysuj ze znaków klawiatury filiżankę kawy albo pierwszą literę swojego imienia. Rysunek ma mieć co najmniej 5 wierszy.

Wskazówka

Zadanie 07 to najlepszy trening: im szybciej rozpoznasz typowe błędy, tym mniej czasu stracisz później.

TYDZIEŃ 01

KONTROLA POSTĘPÓW

Potrafię...

- [ ] zainstalować JDK i uruchomić program w IDE
- [ ] utworzyć klasę z metodą main
- [ ] wypisać tekst za pomocą print i println
- [ ] napisać komentarz jedno- i wielowierszowy
- [ ] odczytać prosty komunikat błędu



Miniprojekt: Karta powitalna

Napisz program, który wypisze ozdobną kartę z ramką z gwiazdek. Na karcie umieść swoje imię, datę rozpoczęcia nauki i cel na najbliższe 12 tygodni.

Dziennik nauki

Czas nauki w tym tygodniu: _____ h

Co poszło najlepiej? _____

Co było trudne? _____

Pytanie do sprawdzenia: _____

Jak oceniasz ten tydzień?

○ ○ ○ ○ ○

1 2 3 4 5

TYDZIEŃ 02

02 WEJŚCIE, WYJŚCIE, ZMIENNE I TYPY

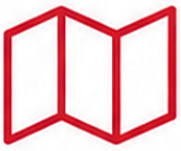
Cele tygodnia

Nauczysz się przechowywać dane w zmiennych, dobierać do nich typy i wczytywać informacje od użytkownika.

Tematy

- Deklaracja i inicjalizacja zmiennych
- Typy proste: int, double, boolean, char
- Tekst w zmiennych: String
- Stałe i słowo final
- Nazwy zmiennych w stylu camelCase
- Wczytywanie danych: Scanner
- Łączenie tekstu ze zmiennymi

Mapy do przerobienia



- mapy o zmiennych i typach danych
- mapy o wejściu i wyjściu danych
- mapy o nazewnictwie i stałych

```java
int wiek = 25;
String imie = "Ola";
System.out.println(imie + " ma " + wiek + " lat");
```

Wskazówka

Tabelę typów znajdziesz w Bonusie #3. Przejrzyj ją przed pierwszą sesją.

5 sesji po 45–60 minut

TYDZIEŃ 02

PLAN DZIEŃ PO DNIU

Dzień 1

60 min

Zmienne i typy proste

Przerób mapy o zmiennych i typach. Zapisz w zeszycie, do czego służy każdy typ.

Dzień 2

60 min

Tekst i stałe

Przerób mapy o typie String i słowie final. Wykonaj zadania 09–10.

Dzień 3

60 min

Wczytywanie danych

Przerób mapy o wejściu danych i klasie Scanner. Wykonaj zadania 11–12.

Dzień 4

60 min

Łączenie danych

Wykonaj zadania 13–14. Sprawdź, co się stanie, gdy wpiszesz literę zamiast liczby.

Dzień 5

60 min

Utrwalenie

Wykonaj zadania 15–16 i wypełnij kontrolę postępów.

Weekend: powtórz mapy o typach danych.
To fundament kolejnych tygodni.



TYDZIEŃ 02

PUŁAPKI I WSKAZÓWKI

```
Scanner sc = new Scanner(System.in);
System.out.print("Podaj wiek: ");
int wiek = sc.nextInt();
sc.nextLine();
System.out.print("Podaj imię: ");
String imie = sc.nextLine();
```

Zgubiony Enter

Po nextInt() w buforze zostaje znak nowej linii. Bez dodatkowego sc.nextLine() kolejne wczytanie tekstu zwróci pusty napis.

Przecinek czy kropka?

W kodzie liczby dziesiętne zapisujesz z kropką: 3.5. Scanner może jednak oczekiwać przecinka, jeśli komputer ma polskie ustawienia regionalne.

Brak importu

Na początku pliku dodaj import java.util.Scanner; bez tego program się nie skompiluje.

Wskazówka

Nadawaj zmiennym nazwy, które mówią, co przechowują: wiek, cena, liczbaUczniow. Nazwy typu x czy a2 utrudniają czytanie kodu.

TYDZIEŃ 02

ZADANIA TYGODNIA (1/2)

Każdy program uruchom kilka razy z różnymi danymi.

09 ZADANIE 09 — Moje dane w zmiennych

Utwórz zmienne na imię, wiek, wzrost w metrach i informację, czy lubisz kawę. Wypisz je w czytelnych zdaniach.

10 ZADANIE 10 — Stała cena

Zadeklaruj stałą final z ceną kawy. Wypisz koszt trzech kaw i spróbuj zmienić wartość stałej. Zanotuj, co się stało.

11 ZADANIE 11 — Powitanie użytkownika

Wczytaj imię użytkownika za **pomocą klasy Scanner** i wypisz spersonalizowane powitanie.

12 ZADANIE 12 — Kalkulator wieku

Wczytaj rok urodzenia i wypisz, ile lat skończy użytkownik w tym roku.

TYDZIEŃ 02

ZADANIA TYGODNIA (2/2)

ZADANIE 13

Formularz rejestracji

Wczytaj imię, nazwisko, miasto i wiek. Wypisz wszystkie dane w formie uporządkowanego formularza.

ZADANIE 14

Paragon za zakupy

Wczytaj nazwę produktu, cenę i liczbę sztuk. Wypisz paragon z kwotą do zapłaty.

ZADANIE 15

Pierwsza litera

Wczytaj słowo i zapisz jego pierwszą literę w zmiennej typu char za pomocą charAt(0). Wypisz ją na ekranie.

ZADANIE 16

Test błędnych danych

Uruchom program z zadania 12 i wpisz tekst zamiast liczby. Zapisz nazwę wyjątku, który się pojawił. Wrócisz do niego w tygodniu 11.

Wskazówka

Błąd w zadaniu 16 jest zamierzony. Na razie wystarczy, że wiesz, skąd się bierze.

TYDZIEŃ 02

KONTROLA POSTĘPÓW



Potrafię...

- [ ] zadeklarować zmienną i nadać jej wartość
- [ ] wczytać liczbę i tekst klasą Scanner
- [ ] dobrać typ do rodzaju danych
- [ ] połączyć tekst ze zmiennymi
- [ ] utworzyć stałą ze słowem final
- [ ] nazwać zmienne w stylu camelCase

Miniprojekt: Ankieta o kawie

Zadaj użytkownikowi pięć pytań: imię, wiek, ulubiona kawa, liczba kaw dziennie i cena jednej kawy. Na koniec wypisz podsumowanie z tygodniowym kosztem kawy.

Jak oceniasz ten tydzień?

1 2 3 4 5

Dziennik nauki

Czas nauki w tym tygodniu: ____ h

Co poszło najlepiej? _____

Co było trudne? _____

Pytanie do sprawdzenia: _____

TYDZIEŃ 03

03

>_ OPERATORY I KONWERSJE

Cele tygodnia

Nauczysz się liczyć w Javie, porównywać wartości i bezpiecznie zamieniać jeden typ danych na inny.

5 sesji po 45–60 minut

Tematy

- Operatory arytmetyczne: + - * / %
- Dzielenie całkowite i reszta z dzielenia
- Inkrementacja i dekrementacja: ++ i --
- Operatory przypisania: += -= *=
- Operatory porównania i logiczne
- Konwersje automatyczne i rzutowanie
- Zamiana tekstu na liczbę: Integer.parseInt

Mapy do przerobienia

- mapy o operatorach
- mapy o konwersjach i rzutowaniu typów
- mapy o wyrażeniach logicznych

Wskazówka

Zanim uruchomisz program, zgadnij wynik. Potem sprawdź, czy miałeś rację.



TYDZIEŃ 03

PLAN DZIEŃ PO DNIU

Dzień 1

Działania na liczbach

Przerób mapy o operatorach arytmetycznych. Oblicz kilka wyrażeń w zeszycie, a potem w kodzie.

60 min

Dzień 2

Reszta z dzielenia

Przećwicz operator % i dzielenie całkowite. Wykonaj zadania 17–18.

60 min

Dzień 3

Porównania i logika

Przerób mapy o wyrażeniach logicznych. Wykonaj zadania 19–20.

60 min

Dzień 4

Konwersje typów

Przerób mapy o konwersjach i rzutowaniu. Wykonaj zadania 21–22.

60 min

Dzień 5

Podsumowanie etapu

Wykonaj zadania 23–24 i wypełnij kontrolę postępów oraz podsumowanie etapu 1.

60 min

Weekend: przejrzyj wszystkie mapy z etapu 1. W przyszłym tygodniu zaczyna się logika.

TYDZIEŃ 03

PUŁAPKI I WSKAZÓWKI

Błąd

```
double srednia = 7 / 2;
```

→

Poprawka

```
double srednia = 7 / 2.0;
```

Dzielenie dwóch liczb całkowitych daje liczbę całkowitą: 3, a nie 3.5.

Błąd

```
if (wynik = 10)
```

→

Poprawka

```
if (wynik == 10)
```

Pojedynczy znak = przypisuje wartość. Do porównania służy ==.

Błąd

```
int x = (int) 9.99;
```

→

Poprawka

```
int x = (int) Math.round(9.99);
```

Rzutowanie obcina część ułamkową i daje 9. Jeśli chcesz zaokrąglić, użyj Math.round.

Wskazówka

W razie wątpliwości dodaj nawiasy, zamiast polegać na kolejności działań. Zapis (a + b) * c czyta się łatwiej.

TYDZIEŃ 03

ZADANIA TYGODNIA (1/2)

Przed uruchomieniem zapisz w komentarzu, jaki wynik przewidujesz.

ZADANIE 17 — Kalkulator dwóch liczb

Wczytaj dwie liczby całkowite i wypisz ich sumę, różnicę, iloczyn, iloraz oraz resztę z dzielenia.

ZADANIE 18 — Parzysta czy nie?

Wczytaj liczbę i za pomocą operatora % wypisz true, jeśli jest parzysta, albo false, jeśli jest nieparzysta.

ZADANIE 19 — Minuty na godziny

Wczytaj liczbę minut i wypisz ją jako godziny i minuty, na przykład 135 minut to 2 h 15 min.

ZADANIE 20 — Pełnoletni z biletem

Wczytaj wiek i informację, czy ktoś ma bilet. Wypisz wynik wyrażenia: wiek większy lub równy 18 oraz posiadanie biletu.

TYDZIEŃ 03

ZADANIA TYGODNIA (2/2)

21 ZADANIE 21 — Tekst na liczbę

Wczytaj liczbę jako tekst metodą nextLine(), zamień ją na int za pomocą Integer.parseInt i wypisz jej podwojoną wartość.

22 ZADANIE 22 — Średnia ocen

Wczytaj trzy oceny całkowite i wypisz ich średnią jako liczbę dziesiętną. Zadbaj, by wynik 4.33 nie zamienił się w 4.

23 ZADANIE 23 — Licznik kroków

Utwórz zmienną kroki równą 0. Zwiększ ją kolejno operatorami ++, += 100 i *= 2. Po każdej zmianie wypisz wartość.

24 ZADANIE 24 — Przelicznik walut ⭐

Wczytaj kwotę w złotych i kurs euro. Wypisz kwotę w euro zaokrągloną do dwóch miejsc po przecinku.

Wskazówka

W zadaniu 24 możesz użyć Math.round(kwota * 100) / 100.0.
Sprawdź w mapach o konwersjach, dlaczego na końcu jest 100.0.

TYDZIEŃ 03

KONTROLA POSTĘPÓW

Potrafię...

- [ ] używać operatorów + - * / %
- [ ] wyjaśnić, czym jest dzielenie całkowite
- [ ] porównywać wartości i łączyć warunki
- [ ] rzutować typy i zamieniać tekst na liczbę

Dziennik nauki

Czas nauki w tym tygodniu: ____ h

Co było trudne? _____

Podsumowanie etapu 1

Za Tobą trzy tygodnie i 24 zadania. Umiesz już uruchomić program, przechowywać dane, wczytywać je od użytkownika i liczyć. To solidne podstawy, na których zbudujesz logikę programów.

Miniprojekt: Kalkulator rachunku

Wczytaj kwotę rachunku w restauracji, procent napiwku i liczbę osób. Wypisz napiwek, sumę i kwotę na osobę.

Jak oceniasz ten tydzień?

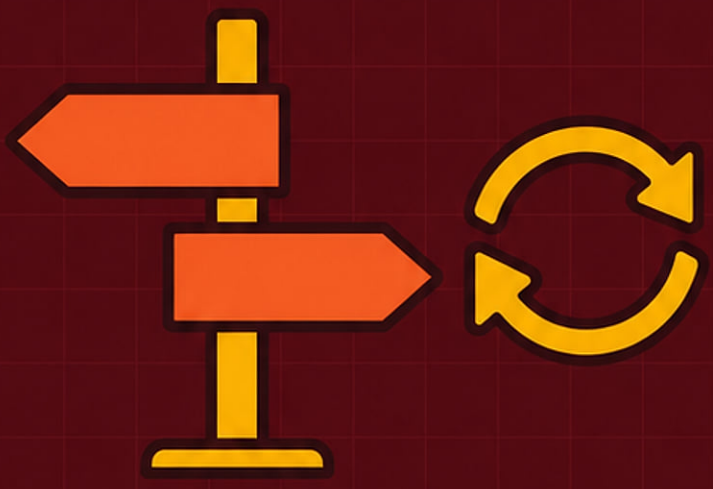
☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5

2

ETAP 2

LOGIKA

Tygodnie 4–6: decyzje w kodzie, pętle i metody



TYDZIEŃ 04

04

DECYZJE W KODZIE

Cele tygodnia

Nauczysz program podejmować decyzje i reagować na różne dane, a także rozbijać problem na małe kroki.

Tematy

- Instrukcja if i blok else
- Wiele warunków: else if
- Warunki złożone: && i ||
- Porównywanie tekstu metodą equals
- Instrukcja switch
- Operator warunkowy ? :
- Rozwiązywanie problemów krok po kroku

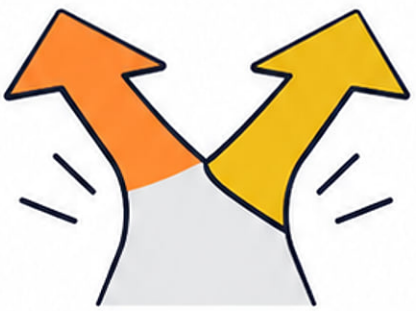
Mapy do przerobienia

- mapy o instrukcjach warunkowych
- mapy o strukturach decyzyjnych i switch
- mapy o rozwiązywaniu problemów

5 sesji po 45–60 minut

Wskazówka

Od tego tygodnia możesz sięgać po Bonus #2 · 50 praktycznych wyzwań z Javy. Wybieraj wyzwania z tematów, które już znasz.



TYDZIEŃ 04

PLAN DZIEŃ PO DNIU

Dzień 1	**Pierwsze decyzje** Przerób mapy o instrukcji if i bloku else. Napisz trzy proste warunki w zeszycie.	60 min
Dzień 2	**Wiele ścieżek** Przećwicz else if i warunki złożone. Wykonaj zadania 25–26.	60 min
Dzień 3	**Instrukcja switch** Przerób mapy o strukturach decyzyjnych. Wykonaj zadania 27–28.	60 min
Dzień 4	**Rozwiązywanie problemów** Przerób mapy o rozwiązywaniu problemów. Wykonaj zadania 29–30.	60 min
Dzień 5	**Utrwalenie** Wykonaj zadania 31–32 i wypełnij kontrolę postępów.	60 min

Weekend: wybierz dwa wyzwania o warunkach z Bonusu #2.

TYDZIEŃ 04

PUŁAPKI I WSKAZÓWKI

```
String haslo = sc.nextLine();
if (haslo.equals("java123")) {
    System.out.println("Witaj!");
} else {
    System.out.println("Złe hasło");
}
```

== przy tekstach

Tekst porównuj metodą equals. Operator == sprawdza, czy to ten sam obiekt, a nie czy napisy są takie same.

Zapomniany break

W klasycznym switch bez break program wykona też kolejne przypadki.

Średnik po if

Zapis `if (x > 0);` kończy instrukcję. Blok w nawiasach klamrowych wykona się zawsze.

Wskazówka

Zanim napiszesz kod, wypisz w punktach wszystkie możliwe przypadki. Każdy punkt to jedna gałąź warunku.

TYDZIEŃ 04

ZADANIA TYGODNIA (1/2)

Testuj każdy program na co najmniej trzech różnych zestawach danych.

ZADANIE 25 — Plus, minus czy zero?

Wczytaj liczbę całkowitą i wypisz, czy jest dodatnia, ujemna, czy równa zero.

ZADANIE 26 — Ocena z punktów

Wczytaj liczbę punktów od 0 do 100 i wypisz ocenę szkolną od 1 do 6 według własnej skali.

ZADANIE 27 — Dzień tygodnia

Wczytaj numer od 1 do 7 i za pomocą switch wypisz nazwę dnia tygodnia. Dla innych liczb wypisz komunikat o błędzie.

ZADANIE 28 — Menu kawiarni

Wypisz menu z trzema napojami. Wczytaj wybór użytkownika i wypisz cenę wybranego napoju.

Sprawdź też wartości graniczne, na przykład 0 i 100 w zadaniu 26.

TYDZIEŃ 04

>_ ZADANIA TYGODNIA (2/2)

29 **ZADANIE 29 — Rok przestępny** ☐

Wczytaj rok i sprawdź, czy jest przestępny: podzielny przez 4, ale nie przez 100, albo podzielny przez 400.

30 **ZADANIE 30 — Logowanie** ☐

Wczytaj login i hasło. Wypisz „Zalogowano", gdy oba są poprawne, a w przeciwnym razie wskaż, co jest błędne.

31 **ZADANIE 31 — Największa z trzech** ☐

Wczytaj trzy liczby i wypisz największą z nich. Najpierw rozpisz rozwiązanie w punktach, a dopiero potem napisz kod.

32 **ZADANIE 32 — Bilet do kina ⭐** ☐

Wczytaj wiek i dzień tygodnia. Oblicz cenę biletu: zniżka dla dzieci i seniorów, a we wtorki tańsze bilety dla wszystkich.

Wskazówka

W zadaniu 32 możesz użyć operatora ? : , aby krótko wybrać jedną z dwóch cen.

TYDZIEŃ 04

KONTROLA POSTĘPÓW

Potrafię...



- [ ] napisać instrukcję if, else if i else
- [ ] łączyć warunki operatorami && i ||
- [ ] porównać teksty metodą *equals*
- [ ] użyć instrukcji *switch*
- [ ] rozpisać problem w punktach przed kodowaniem

Miniprojekt: Quiz o Javie

Przygotuj quiz z pięcioma pytaniami. Po każdej odpowiedzi wypisz, czy jest poprawna, a na końcu pokaż wynik i ocenę słowną, na przykład „Świetnie!" albo „Spróbuj jeszcze raz".

Dziennik nauki

Czas nauki w tym tygodniu: _____ h

Co poszło najlepiej? _____

Co było trudne? _____

Pytanie do sprawdzenia: _____

Jak oceniasz ten tydzień?

1 2 3 4 5

TYDZIEŃ 05

PĘTLE I POWTÓRZENIA 05

Cele tygodnia

Nauczysz się powtarzać fragmenty kodu, liczyć sumy i przerywać pętlę we właściwym momencie.

Tematy

- Pętla for i licznik
- Pętla while
- Pętla do-while
- Instrukcje break i continue
- Suma i licznik w pętli
- Pętle zagnieżdżone
- Jak uniknąć pętli nieskończonej

Mapy do przerobienia

- mapy o pętlach for, while i do-while
- mapy o break i continue
- mapy o iteracjach

```
for (int i = 1; i <= 5; i++) {
    System.out.println("Kawa nr " + i);
}
```

Wskazówka

Na kartce rozpisz wartość licznika w każdym obrocie pętli. To najlepszy sposób, by zrozumieć jej działanie.

5 sesji po 45–60 minut

TYDZIEŃ 05

PLAN DZIEŃ PO DNIU

Dzień 1

60 min

Pętla for

Przerób mapy o pętli for. Wypisz liczby od 1 do 10, a potem od 10 do 1.

Dzień 2

60 min

While i do-while

Przerób mapy o pętlach while i do-while. Wykonaj zadania 33–34.

Dzień 3

60 min

Break i continue

Przerób mapy o break i continue. Wykonaj zadania 35–36.

Dzień 4

60 min

Pętle w pętlach

Przerób mapy o iteracjach. Wykonaj zadania 37–38.

Dzień 5

60 min

Utrwalenie

Wykonaj zadania 39–40 i wypełnij kontrolę postępów.

Weekend: wybierz z Bonusu #2 dwa wyzwania z pętlami.



TYDZIEŃ 05

PUŁAPKI I WSKAZÓWKI

```
int suma = 0;
int i = 1;
while (i <= 10) {
    suma += i;
    i++;
}
System.out.println(suma);
```

Pętla bez końca

Jeśli zapomnisz o i++, warunek zawsze będzie prawdziwy, a program nigdy się nie zatrzyma.

Błąd o jeden

Warunek i < 10 i warunek i <= 10 dają różną liczbę powtórzeń. Zawsze sprawdź pierwszy i ostatni obrót pętli.

Suma w złym miejscu

Zmienną suma utwórz przed pętlą. Utworzona w środku pętli będzie zerowana przy każdym powtórzeniu.

Wskazówka

Gdy program się zawiesi, zatrzymaj go przyciskiem stop w IDE i sprawdź, czy licznik się zmienia.

TYDZIEŃ 05

ZADANIA TYGODNIA (1/2)

Najpierw zapisz, ile razy ma się wykonać pętla.

33 ZADANIE 33 — Tabliczka mnożenia

Wczytaj liczbę od 1 do 10 i za pomocą pętli *for* wypisz jej tabliczkę mnożenia od 1 do 10.

34 ZADANIE 34 — Suma do zera

Wczytuj liczby w pętli while, dopóki użytkownik nie wpisze 0. Na końcu wypisz ich sumę.

35 ZADANIE 35 — Zgadnij liczbę

Program losuje liczbę od 1 do 50 za pomocą Math.random(). Użytkownik zgaduje w pętli do-while, a program podpowiada: za mało albo za dużo.

36 ZADANIE 36 — Bez trzynastki

Wypisz liczby od 1 do 30, ale pomiń 13 za pomocą continue i przerwij pętlę instrukcją break na pierwszej liczbie podzielnej przez 25.

TYDZIEŃ 05

ZADANIA TYGODNIA (2/2)

ZADANIE 37

— Kwadrat z gwiazdek

Wczytaj rozmiar i za pomocą dwóch pętli zagnieżdżonych wypisz kwadrat z gwiazdek o podanym boku.

ZADANIE 38

— Choinka

Wypisz trójkąt z gwiazdek o wysokości 5 wierszy, w którym każdy wiersz ma o dwie gwiazdki więcej.

ZADANIE 39

— Silnia

Wczytaj liczbę od 1 do 10 i oblicz jej silnię w pętli. Wypisz cały zapis, na przykład 4! = 24.

ZADANIE 40

— Bankomat

Saldo wynosi 1000 zł. W pętli pokazuj menu: sprawdź saldo, wypłać, wpłać, zakończ. Nie pozwól wypłacić więcej, niż jest na koncie.

Wskazówka

W zadaniu 38 zapisz najpierw, ile spacji i ile gwiazdek ma każdy wiersz.

TYDZIEŃ 05

KONTROLA POSTĘPÓW



Potrafię...

- [ ] napisać pętlę for z licznikiem
- [ ] dobrać pętlę while lub do-while
- [ ] użyć break i continue
- [ ] policzyć sumę w pętli
- [ ] zbudować pętlę zagnieżdżoną
- [ ] rozpoznać i naprawić pętlę bez końca

Miniprojekt: Trener tabliczki mnożenia

Program zadaje dziesięć losowych pytań z tabliczki mnożenia, liczy poprawne odpowiedzi i na końcu wypisuje wynik. Po zakończeniu pyta, czy zagrać jeszcze raz.

Jak oceniasz ten tydzień?

1 2 3 4 5

Dziennik nauki

Czas nauki w tym tygodniu: _____ h

Co poszło najlepiej? _____

Co było trudne? _____

Pytanie do sprawdzenia: _____

TYDZIEŃ 06

06

METODY I PORZĄDEK W KODZIE

Cele tygodnia

Nauczysz się dzielić program na małe metody, przekazywać im dane i odbierać wyniki.

5 sesji po 45–60 minut

Tematy

- Budowa metody i słowo void
- Parametry i argumenty
- Zwracanie wartości: return
- Przeciążanie metod
- Zasięg zmiennych lokalnych
- Pakiety i instrukcja import
- Jedna metoda, jedno zadanie

Mapy do przerobienia

- mapy o metodach i parametrach
- mapy o zwracaniu wartości i przeciążaniu
- mapy o organizacji kodu w pakiety

Wskazówka

Nazwę metody zacznij od czasownika: policzSume, wypiszMenu, sprawdzHaslo. Od razu widać, co robi.



TYDZIEŃ 06

PLAN DZIEŃ PO DNIU

Dzień 1

Pierwsza metoda

Przerób mapy o budowie metody. Przenieś powitanie z tygodnia 1 do osobnej metody.

60 min

Dzień 2

Parametry i return

Przerób mapy o parametrach i zwracaniu wartości. Wykonaj zadania 41–42.

60 min

Dzień 3

Przeciążanie i zasięg

Przerób mapy o przeciążaniu metod. Wykonaj zadania 43–44.

60 min

Dzień 4

Pakiety

Przerób mapy o organizacji kodu w pakiety. Wykonaj zadania 45–46.

60 min

Dzień 5

Podsumowanie etapu

Wykonaj zadania 47–48 i wypełnij kontrolę postępów z podsumowaniem etapu 2.

60 min

Weekend: przejrzyj wszystkie mapy z etapu 2. Od przyszłego tygodnia pracujesz na większych zbiorach danych.

TYDZIEŃ 06

PUŁAPKI I WSKAZÓWKI

Błąd

```
static void suma(int a, int b) {
   int wynik = a + b;
}
```

Poprawka

```
static int suma(int a, int b) {
   return a + b;
}
```

Metoda void niczego nie zwraca. Jeśli wynik ma trafić dalej, podaj typ i użyj *return*.

Błąd

```
suma(2, 3);
```

Poprawka

```
int s = suma(2, 3);
```

Zwrócona wartość przepada, jeśli jej nie zapiszesz ani nie wypiszesz.

Błąd

Zmienna z wnętrza metody użyta w main.

Poprawka

Przekaż wartość przez return.

Zmienna lokalna istnieje tylko w bloku, w którym ją utworzono.

Wskazówka

Jeśli metoda ma więcej niż 20 linii, prawdopodobnie robi kilka rzeczy naraz. Podziel ją na mniejsze.

TYDZIEŃ 06

ZADANIA TYGODNIA (1/2)

Każdą metodę przetestuj, wywołując ją co najmniej dwa razy z main.

41 ☐

ZADANIE 41 — Metoda powitalna

Napisz metodę przywitaj, która przyjmuje imię i wypisuje powitanie. Wywołaj ją dla trzech różnych imion.

42 ☐

ZADANIE 42 — Pole prostokąta

Napisz metodę, która przyjmuje długość i szerokość, a zwraca pole prostokąta. Wynik wypisz w main.

43 ☐

ZADANIE 43 — Czy parzysta?

Napisz metodę czyParzysta, która przyjmuje liczbę i zwraca true albo false. Użyj jej w pętli od 1 do 10.

44 ☐

ZADANIE 44 — Przeciążone dodawanie

Napisz trzy wersje metody dodaj: dla dwóch liczb int, dla trzech liczb int i dla dwóch liczb double.

Metoda, która zwraca wynik, jest łatwiejsza do sprawdzenia niż metoda, która tylko wypisuje.

TYDZIEŃ 06

ZADANIA TYGODNIA (2/2)

45 ZADANIE 45 — Własny pakiet

Utwórz pakiet narzedzia z klasą Kalkulator i metodami do czterech działań. Zaimportuj ją w klasie Main i użyj jej metod.

46 ZADANIE 46 — Przelicznik temperatur

Napisz dwie metody: z Celsjusza na Fahrenheita i z Fahrenheita na Celsjusza. Wypisz tabelę dla temperatur od 0 do 40 co 10 stopni.

47 ZADANIE 47 — Porządki w kodzie

Weź program bankomatu z zadania 40 i podziel go na metody: pokazMenu, wplac, wyplac i pokazSaldo.

48 ZADANIE 48 — Mocne hasło ⭐

Napisz metodę, która sprawdza, czy hasło ma co najmniej 8 znaków i zawiera cyfrę. Zwróć true albo false.

Wskazówka

Zadanie 47 pokazuje, po co są **metody**: ten sam program staje się krótszy w main i dużo łatwiejszy w czytaniu.

TYDZIEŃ 06

KONTROLA POSTĘPÓW

Potrafię...

- [ ] napisać metodę z parametrami
- [ ] zwrócić wynik za pomocą return
- [ ] przeciążyć metodę
- [ ] wyjaśnić, czym jest zasięg zmiennej
- [ ] podzielić program na pakiety

Dziennik nauki

Czas nauki w tym tygodniu: _____ h

Co było trudne? _____

_____

_____

_____



Podsumowanie etapu 2

Za Tobą sześć tygodni i 48 zadań. Twoje programy podejmują decyzje, powtarzają działania i są podzielone na metody. To już prawdziwa logika programowania.

Miniprojekt: Gra w kości

Dwóch graczy rzuca na zmianę kostką, aż jeden z nich zdobędzie 30 punktów. Rzut, sprawdzanie wyniku i wypisywanie stanu gry umieść w osobnych metodach.

Jak oceniasz ten tydzień?

○ ○ ○ ○ ○

1 2 3 4 5

3

ETAP 3

DANE I OBIEKTY

Tygodnie 7–9: tablice, Stringi, kolekcje oraz klasy i obiekty



TYDZIEŃ 07

07

TABLICE I STRINGI

Cele tygodnia

Nauczysz się przechowywać wiele wartości w jednej tablicy i swobodnie pracować z tekstem.

Tematy

- Tworzenie tablicy i indeksy od zera
- Długość tablicy: length
- Przechodzenie po tablicy: for i for-each
- Wyznaczanie minimum, maksimum i sumy
- Metody klasy String: length, charAt, substring, indexOf
- Dzielenie tekstu: split
- Łączenie tekstu: StringBuilder

Mapy do przerobienia

- mapy o tablicach
- mapy o Stringach i ich metodach
- mapy o przetwarzaniu tekstu

5 sesji po 45–60 minut

Wskazówka

Rysuj tablice na kartce jako rząd pudełek z numerami od 0. Wtedy indeksy przestają być abstrakcyjne.



TYDZIEŃ 07

PLAN DZIEŃ PO DNIU

Dzień 1

Pierwsza tablica

Przerób mapy o tablicach. Utwórz tablicę z ocenami i wypisz każdy jej element.

60 min

Dzień 2

Obliczenia na tablicach

Przećwicz pętlę for-each, sumowanie i szukanie maksimum. Wykonaj zadania 49–50.

60 min

Dzień 3

Metody Stringów

Przerób mapy o Stringach. Wykonaj zadania 51–52.

60 min

Dzień 4

Przetwarzanie tekstu

Przerób mapy o split i StringBuilder. Wykonaj zadania 53–54.

60 min

Dzień 5

Utrwalenie

Wykonaj zadania 55–56 i wypełnij kontrolę postępów.

60 min

Weekend: wybierz z Bonusu #2 dwa wyzwania o tablicach lub tekście.

TYDZIEŃ 07

PUŁAPKI I WSKAZÓWKI

```
int[] oceny = {5, 4, 3, 6};
for (int i = 0; i < oceny.length; i++) {
    System.out.println(oceny[i]);
}
```

Indeks poza tablicą

Tablica z 4 elementami ma indeksy od 0 do 3. Odwołanie do indeksu 4 kończy się wyjątkiem ArrayIndexOutOfBounds Exception.

Tekst się nie zmienia

Metoda toUpperCase() nie zmienia napisu, tylko zwraca nowy. Zapisz wynik:
imie = imie.toUpperCase();

Łączenie w pętli

Doklejanie tekstu przez + w długiej pętli tworzy wiele nowych obiektów. Użyj klasy StringBuilder.

Wskazówka

Do szybkiego podglądu całej tablicy użyj Arrays.toString(oceny).
Nie musisz pisać pętli tylko po to, by sprawdzić zawartość.

TYDZIEŃ 07

ZADANIA TYGODNIA (1/2)

Zanim napiszesz pętlę, narysuj tablicę i podpisz indeksy.

ZADANIE 49

Temperatury tygodnia

Utwórz tablicę z siedmioma temperaturami. Wypisz najniższą, najwyższą i średnią temperaturę tygodnia.

ZADANIE 50

Odwrócona kolejność

Wczytaj pięć liczb do tablicy i wypisz je w odwrotnej kolejności, od ostatniej do pierwszej.

ZADANIE 51

Analiza imienia

Wczytaj imię i wypisz jego długość, pierwszą i ostatnią literę oraz wersję zapisaną wielkimi literami.

ZADANIE 52

Palindrom

Sprawdź, czy wczytane słowo czyta się tak samo od początku i od końca, na przykład „kajak”.

W zadaniu 52 porównuj znaki z obu końców słowa, przesuwając się do środka.

TYDZIEŃ 07

ZADANIA TYGODNIA (2/2)

53 ZADANIE 53 — Licznik samogłosek

Wczytaj zdanie i policz, ile ma samogłosek. Uwzględnij polskie litery: a, ą, e, ę, i, o, ó, u, y.

54 ZADANIE 54 — Słowa w zdaniu

Podziel zdanie metodą split(" ") i wypisz każde słowo w osobnym wierszu razem z jego długością.

55 ZADANIE 55 — Inicjały

Wczytaj imię i nazwisko, a następnie wypisz inicjały z kropkami, na przykład A.K.

56 ZADANIE 56 — Szyfr Cezara

Zaszyfruj wczytane słowo, przesuwając każdą literę o 3 miejsca w alfabecie. Wynik zbuduj za pomocą klasy StringBuilder.

Wskazówka

W zadaniu 56 zacznij od samych małych liter angielskiego alfabetu. Polskie znaki możesz dodać na końcu.

TYDZIEŃ 07

>_ KONTROLA POSTĘPÓW

Potrafię...

- [ ] utworzyć tablicę i odczytać jej elementy
- [ ] przejść po tablicy pętlą for i for-each
- [ ] znaleźć minimum, maksimum i sumę
- [ ] używać metod length, charAt i substring
- [ ] podzielić tekst metodą split
- [ ] budować tekst za pomocą klasy StringBuilder



Miniprojekt: Dziennik ocen

Wczytaj imię ucznia i jego pięć ocen. Wypisz wszystkie oceny, średnią, najlepszą i najsłabszą ocenę oraz informację, czy średnia przekracza 4,0.

Dziennik nauki

Czas nauki w tym tygodniu: ____ h

Co poszło najlepiej? _____

Co było trudne? _____

Pytanie do sprawdzenia: _____

Jak oceniasz ten tydzień?

○ ○ ○ ○ ○

1 2 3 4 5

TYDZIEŃ 08

08

KOLEKCJE I PRZETWARZANIE DANYCH

Cele tygodnia

Nauczysz się przechowywać dane w listach, które rosną i maleją, a także sortować je i przeszukiwać.

Tematy

- ArrayList: add, get, set, remove, size
- Typy opakowujące: Integer, Double
- Przechodzenie po liście
- Sprawdzanie zawartości: contains, indexOf
- Klasa Collections: sort, max, min, reverse
- Pary klucz–wartość: HashMap
- Zliczanie i filtrowanie danych

Mapy do przerobienia

mapy o ArrayList

mapy o Collections

mapy o przetwarzaniu danych

```
ArrayList<String> zakupy = new ArrayList<>();
zakupy.add("kawa");
zakupy.add("mleko");
System.out.println(zakupy.size());
```

Wskazówka

Gdy nie wiesz z góry, ile będzie elementów, wybierz ArrayList zamiast tablicy.

5 sesji po 45–60 minut

TYDZIEŃ 08

>_ PLAN DZIEŃ PO DNIU

Dzień 1 — 60 min
Pierwsza lista

Przerób mapy o ArrayList. Zbuduj listę zakupów, dodaj i usuń kilka elementów.

Dzień 2 — 60 min
Przeszukiwanie listy

Przećwicz contains, indexOf i pętlę po liście. Wykonaj zadania 57–58.

Dzień 3 — 60 min
Sortowanie

Przerób mapy o Collections. Wykonaj zadania 59–60.

Dzień 4 — 60 min
Klucz i wartość

Przerób mapy o przetwarzaniu danych i HashMap. Wykonaj zadania 61–62.

Dzień 5 — 60 min
Utrwalenie

Wykonaj zadania 63–64 i wypełnij kontrolę postępów.

Weekend: przepisz dziennik ocen z tygodnia 7 tak, aby używał ArrayList.



TYDZIEŃ 08

PUŁAPKI I WSKAZÓWKI

```
ArrayList<Integer> liczby = new ArrayList<>();
liczby.add(10);
liczby.add(20);
liczby.add(30);
Collections.sort(liczby);
System.out.println(liczby.get(0));
```

int zamiast Integer

Lista nie przechowuje typów prostych. Zamiast ArrayList<int> napisz ArrayList<Integer>.

Usuwanie w pętli for-each

Usuwanie elementów w trakcie pętli for-each kończy się wyjątkiem. Użyj zwykłej pętli od końca albo metody removeIf.

size czy length?

Tablica ma pole length, String ma metodę length(), a lista ma metodę size().

Wskazówka

Wypisuj całą listę poleceniem System.out.println(lista). Od razu zobaczysz wszystkie elementy w nawiasach kwadratowych.

TYDZIEŃ 08

ZADANIA TYGODNIA (1/2)

Po każdej zmianie listy wypisz ją w całości i sprawdź wynik.

ZADANIE 57 — Lista zakupów

Wczytuj produkty, dopóki użytkownik nie wpisze „koniec”.
Potem wypisz listę i liczbę produktów.

ZADANIE 58 — Czy jest na liście?

Do listy zakupów z zadania 57 dodaj wyszukiwanie: wczytaj nazwę i wypisz, czy produkt jest na liście, a jeśli tak, na której pozycji.

ZADANIE 59 — Ranking wyników

Wczytaj dziesięć wyników punktowych do listy, posortuj je i wypisz trzy najlepsze.

ZADANIE 60 — Bez duplikatów

Utwórz listę imion z powtórzeniami. Zbuduj nową listę, w której każde imię występuje tylko raz.

TYDZIEŃ 08

ZADANIA TYGODNIA (2/2)

ZADANIE 61

Słownik polsko-angielski

Utwórz HashMap z dziesięcioma słowami. Wczytaj polskie słowo i wypisz jego tłumaczenie albo komunikat, że słowa nie ma w słowniku.

ZADANIE 62

Licznik słów

Wczytaj zdanie i za pomocą HashMap policz, ile razy występuje w nim każde słowo.

ZADANIE 63

Filtr cen

Utwórz listę dziesięciu cen. Zbuduj nową listę z cenami niższymi niż 50 zł i wypisz ich sumę.

ZADANIE 64

Magazyn sklepu

Przechowuj w HashMap nazwy produktów i liczbę sztuk. Dodaj menu: przyjmij towar, sprzedaj, pokaż stan, zakończ.

Wskazówka

W zadaniu 62 przydaje się metoda getOrDefault: zwraca 0, gdy słowa jeszcze nie ma w mapie.

TYDZIEŃ 08

KONTROLA POSTĘPÓW



Potrafię...

- [ ] dodawać, odczytywać i usuwać elementy listy
- [ ] sortować dane klasą Collections
- [ ] dobrać typ opakowujący, na przykład Integer
- [ ] zapisać pary klucz–wartość w HashMap
- [ ] przeszukać listę metodami contains i indexOf
- [ ] policzyć i odfiltrować dane

Miniprojekt: Lista zadań

Program przechowuje Twoje zadania na dziś. Pozwala dodać zadanie, oznaczyć je jako wykonane, usunąć je i wypisać listę posortowaną alfabetycznie.

Jak oceniasz ten tydzień?

1 2 3 4 5

Dziennik nauki

Czas nauki w tym tygodniu: ____ h

Co poszło najlepiej? _____

Co było trudne? _____

Pytanie do sprawdzenia: _____

TYDZIEŃ 09

>_ KLASY I OBIEKTY 09

Cele tygodnia

Zaczniesz programować obiektowo: zaprojektujesz własne klasy, utworzysz z nich obiekty i ochronisz ich dane.

5 sesji po 45–60 minut

Tematy

- Klasa jako szablon, obiekt jako egzemplarz
- Pola i metody obiektu
- Tworzenie obiektu: new
- Konstruktor i słowo this
- Enkapsulacja: modyfikator private
- Gettery i settery
- Metoda toString

Mapy do przerobienia

- mapy o klasach i obiektach
- mapy o konstruktorach
- mapy o enkapsulacji

Wskazówka

W tym i w kolejnym tygodniu korzystaj z Bonusu #5 · Wizualny przewodnik po programowaniu obiektowym. Rysunki pomogą Ci zobaczyć, jak obiekty współpracują.



TYDZIEŃ 09

PLAN DZIEŃ PO DNIU

Dzień 1

Pierwsza klasa

Przerób mapy o klasach i obiektach. Opisz w zeszycie trzy przedmioty z biurka jako klasy z polami.

60 min

Dzień 2

Konstruktory

Przerób mapy o konstruktorach i słowie this. Wykonaj zadania 65–66.

60 min

Dzień 3

Enkapsulacja

Przerób mapy o enkapsulacji. Wykonaj zadania 67–68.

60 min

Dzień 4

Obiekty w listach

Połącz klasy z ArrayList z tygodnia 8. Wykonaj zadania 69–70.

60 min

Dzień 5

Podsumowanie etapu

Wykonaj zadania 71–72 i wypełnij kontrolę postępów z podsumowaniem etapu 3.

60 min

Weekend: przejrzyj Bonus #5 i wszystkie mapy z etapu 3.

TYDZIEŃ 09

⚠ PUŁAPKI I WSKAZÓWKI

Błąd

```
Uczen(String imie) {
    imie = imie;
}
```

→

Poprawka

```
Uczen(String imie) {
    this.imie = imie;
}
```

Bez this parametr przypisuje wartość sam sobie, a pole obiektu zostaje puste.

Błąd

```
public double saldo;
```

→

Poprawka

```
private double saldo;
```

Publiczne pole każdy może zmienić. Prywatne pole zmieniasz tylko przez metody, które sprawdzają dane.

Błąd

```
System.out.println(uczen);
```

→

Poprawka

Dodaj w klasie metodę `toString()`.

Bez toString zobaczysz tylko nazwę klasy i dziwny kod zamiast danych obiektu.

Wskazówka

Nazwy klas zaczynaj wielką literą: Uczen, Konto, Produkt. Nazwy obiektów i zmiennych zaczynaj małą.

TYDZIEŃ 09

ZADANIA TYGODNIA (1/2)

Każdą klasę zapisz w osobnym pliku o tej samej nazwie.

65

ZADANIE 65 — Klasa Ksiazka

Utwórz klasę z polami tytuł, autor i liczba stron. Utwórz trzy obiekty i wypisz ich dane.

66

ZADANIE 66 — Konstruktor z danymi

Dodaj do klasy Ksiazka konstruktor, który od razu ustawia wszystkie pola. Porównaj kod przed zmianą i po niej.

67

ZADANIE 67 — Bezpieczne konto

Utwórz klasę Konto z prywatnym saldem i metodami wplac, wyplac i getSaldo. Nie pozwól na ujemną kwotę ani na debet.

68

ZADANIE 68 — Własny toString

Dodaj metodę toString do klas Ksiazka i Konto, aby zwracała czytelne opisy.

W zadaniu **67** metoda wyplac może zwracać **true** albo **false**, w zależności od tego, czy operacja się udała.

TYDZIEŃ 09

ZADANIA TYGODNIA (2/2)

69 ZADANIE 69 — Klasa z listą

Utwórz klasę Biblioteka, która przechowuje obiekty Ksiazka w ArrayList. Dodaj metody dodajKsiazke i wypiszWszystkie.

70 ZADANIE 70 — Wyszukiwarka

Dodaj do klasy Biblioteka metodę, która wyszukuje książki po nazwisku autora i zwraca listę wyników.

71 ZADANIE 71 — Licznik obiektów

Dodaj do klasy Ksiazka pole static, które zlicza utworzone książki. Wypisz wynik po utworzeniu pięciu obiektów.

72 ZADANIE 72 — Ekspres do kawy ★

Zaprojektuj klasę EkspresDoKawy z poziomem wody i ziaren. Metoda zrobKawe zużywa zapasy albo informuje, czego brakuje.

Wskazówka

Zanim zaczniesz zadanie 72, narysuj klasę jako prostokąt: na górze nazwa, w środku pola, na dole metody.

TYDZIEŃ 09

KONTROLA POSTĘPÓW

Potrafię...

- [ ] zaprojektować klasę z polami i metodami
- [ ] utworzyć obiekt za pomocą konstruktora
- [ ] użyć słowa this
- [ ] ukryć dane modyfikatorem private
- [ ] napisać gettery, settery i toString

Dziennik nauki

Czas nauki w tym tygodniu: _____ h

Co było trudne? _____



Podsumowanie etapu 3

Za Tobą dziewięć tygodni i 72 zadania. Przechowujesz dane w tablicach, listach i mapach, a programy budujesz z własnych klas. Zostały Ci tematy zaawansowane i pierwszy projekt.

Miniprojekt: Wypożyczalnia filmów

Utwórz klasy Film i Wypozyczalnia. Program pozwala dodać film, wypożyczyć go, zwrócić i wypisać listę dostępnych tytułów.

Jak oceniasz ten tydzień?

1 2 3 4 5

4

ETAP 4

TEMATY ZAAWANSOWANE I PROJEKTY

Tygodnie 10–12: dziedziczenie, interfejsy, wyjątki, Streamy i Twój pierwszy projekt



TYDZIEŃ 10

10

DZIEDZICZENIE, POLIMORFIZM I INTERFEJSY

Cele tygodnia

Nauczysz się budować rodziny klas, które mają wspólny kod, ale każda z nich zachowuje się po swojemu.

Tematy

- Dziedziczenie: extends
- Konstruktor klasy bazowej: super
- Nadpisywanie metod: @Override
- Polimorfizm: jedna zmienna, wiele form
- Klasy abstrakcyjne
- Interfejsy i słowo implements
- Kiedy dziedziczenie, a kiedy interfejs

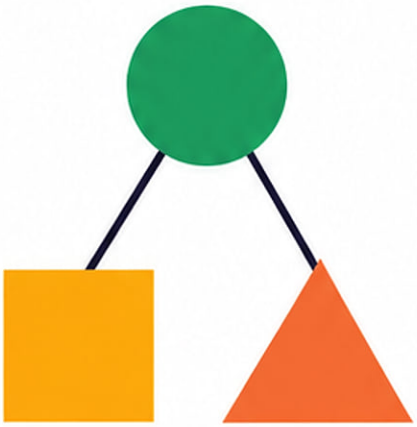
Mapy do przerobienia

mapy o dziedziczeniu

mapy o polimorfizmie

mapy o interfejsach

5 sesji po 45–60 minut



Wskazówka

Przed każdą sesją zajrzyj do Bonusu #5. Prosty test: jeśli zdanie „Pies jest Zwierzęciem” brzmi naturalnie, dziedziczenie ma sens.

TYDZIEŃ 10

PLAN DZIEŃ PO DNIU

Dzień 1	**Dziedziczenie** Przerób mapy o dziedziczeniu. Zbuduj klasę Zwierze i dwie klasy, które po niej dziedziczą.	60 min
Dzień 2	**super i @Override** Przećwicz konstruktor z super i nadpisywanie metod. Wykonaj zadania 73–74.	60 min
Dzień 3	**Polimorfizm** Przerób mapy o polimorfizmie. Wykonaj zadania 75–76.	60 min
Dzień 4	**Interfejsy** Przerób mapy o interfejsach i klasach abstrakcyjnych. Wykonaj zadania 77–78.	60 min
Dzień 5	**Utrwalenie** Wykonaj zadania 79–80 i wypełnij kontrolę postępów.	60 min

Weekend: narysuj schemat klas z zadania 80 tak, jak w Bonusie #5.

TYDZIEŃ 10

PUŁAPKI I WSKAZÓWKI

```
class Pies extends Zwierze {
    @Override
    void dajGlos() {
        System.out.println("Hau!");
    }
}
```

Literówka w nazwie

Bez @Override literówka w nazwie metody tworzy nową metodę zamiast nadpisać starą. Adnotacja wychwyci ten błąd.

super na końcu

Wywołanie super(imie) musi być pierwszą instrukcją w konstruktorze klasy pochodnej.

Wszystko dziedziczy

Dziedzicz tylko wtedy, gdy klasa naprawdę jest rodzajem klasy bazowej. Do wspólnych umiejętności lepiej użyć interfejsu.

Wskazówka

Dzięki polimorfizmowi możesz przechowywać psy i koty w jednej liście typu ArrayList<Zwierze> i wywołać dajGlos() w jednej pętli.

TYDZIEŃ 10

ZADANIA TYGODNIA (1/2)

Zanim zaczniesz pisać, narysuj, która klasa dziedziczy po której.

ZADANIE 73 — Rodzina zwierząt

Utwórz klasę Zwierze z polem imie oraz klasy Pies i Kot, które po niej dziedziczą i mają własną metodę dajGlos.

ZADANIE 74 — Konstruktor z super

Dodaj do klas z zadania 73 konstruktory, które przekazują imię do klasy bazowej przez super.

ZADANIE 75 — Schronisko

Umieść pięć różnych zwierząt w jednej liście ArrayList<Zwierze> i w jednej pętli wywołaj dla każdego dajGlos().

ZADANIE 76 — Pracownicy

Utwórz klasę Pracownik z metodą obliczPensje oraz klasy Kierownik i Stazysta, które liczą pensję na różne sposoby.

Jeśli w zadaniu 75 każde zwierzę mówi po swojemu, polimorfizm działa.

TYDZIEŃ 10

ZADANIA TYGODNIA (2/2)

77

ZADANIE 77 — Figury geometryczne

Utwórz klasę abstrakcyjną Figura z metodą pole() oraz klasy Kolo, Prostokat i Trojkat. Wypisz pola wszystkich figur z jednej listy.

78

ZADANIE 78 — Interfejs Platnosc

Utwórz interfejs z metodą zaplac(kwota) i dwie klasy: PlatnoscKarta i PlatnoscBlik. Każda wypisuje inny komunikat.

79

ZADANIE 79 — Porównywalne produkty

Zaimplementuj w klasie Produkt interfejs Comparable, aby listę produktów dało się posortować według ceny.

80

ZADANIE 80 — Garaż

Zaprojektuj klasy Pojazd, Samochod i Rower oraz interfejs Elektryczny. Utwórz garaż z listą pojazdów i wypisz, które z nich są elektryczne.

Wskazówka

W zadaniu 80 sprawdź typ obiektu operatorem instanceof.

TYDZIEŃ 10

KONTROLA POSTĘPÓW

Potrafię...

- [ ] utworzyć klasę pochodną za pomocą extends
- [ ] wywołać konstruktor bazowy przez super
- [ ] nadpisać metodę z adnotacją @Override
- [ ] wykorzystać polimorfizm w liście obiektów
- [ ] napisać klasę abstrakcyjną i interfejs



Miniprojekt: Arena bohaterów

Utwórz klasę Bohater oraz klasy Wojownik, Mag i Lucznik, które atakują na różne sposoby. Dwóch bohaterów walczy w turach, aż jeden z nich straci wszystkie punkty życia.

Dziennik nauki

Czas nauki w tym tygodniu: _____ h

Co poszło najlepiej?

_____

Co było trudne?

_____

Pytanie do sprawdzenia:

_____

Jak oceniasz ten tydzień?

1 2 3 4 5

TYDZIEŃ 11

11

WYJĄTKI, PLIKI, GENERYKI, LAMBDY I STREAMY

Cele tygodnia

Nauczysz się obsługiwać błędy, zapisywać dane w plikach i zwięźle przetwarzać kolekcje za pomocą lambd i Streamów.

Tematy

- Wyjątki: try, catch, finally
- Rzucanie wyjątków: throw
- Odczyt i zapis plików tekstowych
- Typy generyczne
- Wyrażenia lambda
- Streamy: filter, map, collect
- Korzystanie z dokumentacji API

Mapy do przerobienia

- mapy o wyjątkach
- mapy o plikach
- mapy o generykach i lambdach
- mapy o Streamach i API

```
try {
    int x = Integer.parseInt(tekst);
} catch (NumberFormatException e) {
    System.out.println("To nie liczba!");
}
```

Wskazówka

To najintensywniejszy tydzień planu. Jeśli trzeba, rozłóż go na dwa.

5 sesji po 60 minut

TYDZIEŃ 11

PLAN DZIEŃ PO DNIU

Dzień 1

60 min

Wyjątki

Przerób mapy o wyjątkach. Wróć do błędu z zadania 16 i obsłuż go. Wykonaj zadania 81–82.

Dzień 2

60 min

Pliki

Przerób mapy o plikach. Wykonaj zadania 83–84.

Dzień 3

60 min

Generyki

Przerób mapy o typach generycznych. Wykonaj zadanie 85.

Dzień 4

60 min

Lambdy

Przerób mapy o wyrażeniach lambda. Wykonaj zadanie 86.

Dzień 5

60 min

Streamy i API

Przerób mapy o Streamach i API. Wykonaj zadania 87–88 i wypełnij kontrolę postępów.

Weekend: wybierz projekt na tydzień 12 i przejrzyj Bonus #4 · 10 projektów w Javie od zera.



TYDZIEŃ 11

PUŁAPKI I WSKAZÓWKI

```
List<Integer> oceny = List.of(5, 3, 6, 4);
List<Integer> dobre = oceny.stream()
    .filter(o -> o >= 4)
    .collect(Collectors.toList());
System.out.println(dobre);
```

Pusty catch

Blok catch bez żadnej instrukcji ukrywa błąd. Wypisz przynajmniej komunikat, aby wiedzieć, co się stało.

Niezamknięty plik

Otwarty plik trzeba zamknąć. Najprościej użyć konstrukcji try z zasobami, która zamyka plik automatycznie.

Stream bez końca

Same operacje filter i map niczego nie wykonują. Stream działa dopiero po operacji kończącej, na przykład collect lub forEach.

Wskazówka

Nie znasz metody? Wyszukaj w dokumentacji API nazwę klasy, na przykład „Java Files", i przeczytaj przykłady.

TYDZIEŃ 11

ZADANIA TYGODNIA (1/2)

Przy każdym zadaniu sprawdź, co się stanie, gdy coś pójdzie nie tak.

81 ZADANIE 81 — Bezpieczne wczytywanie

Wczytuj liczbę w pętli, dopóki użytkownik nie poda poprawnej wartości. Błędy obsłuż za pomocą try i catch.

82 ZADANIE 82 — Własny komunikat błędu

W klasie Konto z zadania 67 niech metoda wyplac rzuca IllegalArgumentException, gdy kwota przekracza saldo. Obsłuż wyjątek w main.

83 ZADANIE 83 — Zapis do pliku

Zapisz listę zakupów z zadania 57 do pliku zakupy.txt, każdy produkt w osobnej linii.

84 ZADANIE 84 — Odczyt z pliku

Wczytaj plik zakupy.txt i wypisz jego linie z numerami. Gdy pliku nie ma, wypisz zrozumiały komunikat.

TYDZIEŃ 11

ZADANIA TYGODNIA (2/2)

ZADANIE 85 ☐

Pudełko na wszystko

Utwórz klasę generyczną Pudelko<T> z metodami wloz i wyjmij. Użyj jej raz dla tekstu, a raz dla liczb.

ZADANIE 86 ☐

Sortowanie lambdą

Posortuj listę imion według długości, a potem alfabetycznie. Porównanie zapisz jako wyrażenie lambda.

ZADANIE 87 ☐

Stream ocen

Z listy ocen wybierz te większe lub równe 4, pomnóż każdą przez 10 i zbierz wyniki do nowej listy.

ZADANIE 88 ★ ☐

Statystyki z pliku

Wczytaj z pliku listę cen. Za pomocą Streamów oblicz ich sumę, średnią i największą wartość.



Wskazówka

W zadaniu 88 przydadzą się metody mapToDouble, sum, average i max.

TYDZIEŃ 11

KONTROLA POSTĘPÓW



Potrafię...

- [ ] obsłużyć wyjątek blokiem try i catch
- [ ] rzucić własny wyjątek słowem throw
- [ ] zapisać dane do pliku i je odczytać
- [ ] napisać prostą klasę generyczną
- [ ] użyć wyrażenia lambda
- [ ] przetworzyć listę za pomocą Streamu

Miniprojekt: Notatnik w konsoli

Program pozwala dodawać notatki, zapisuje je do pliku i wczytuje przy kolejnym uruchomieniu. Notatki z wybranym słowem wyszukuje za pomocą Streamu, a błędne dane obsługuje bez przerywania pracy.

Jak oceniasz ten tydzień?

1 2 3 4 5

Dziennik nauki

Czas nauki w tym tygodniu: ____ h

Co poszło najlepiej? _____

Co było trudne? _____

Pytanie do sprawdzenia: _____

TYDZIEŃ 12

12

POWTÓRKA I PIERWSZY WŁASNY PROJEKT

Cele tygodnia

Powtórzysz najważniejsze tematy i zbudujesz od zera swój pierwszy kompletny program.

5 sesji po 60–90 minut

Tematy

- Powtórka tematów z najniższą oceną
- Planowanie projektu na karcie
- Podział programu na klasy
- Menu i główna pętla programu
- Zapis danych do pliku
- Obsługa błędów i testowanie

Mapy do przerobienia

- mapy z tematów ocenionych najniżej
- mapy o klasach i kolekcjach
- mapy o plikach i wyjątkach

Projekt tygodnia: Menedżer wydatków

Program w konsoli, który zapisuje Twoje wydatki, dzieli je na kategorie, liczy statystyki i przechowuje dane w pliku. Więcej pomysłów znajdziesz w Bonusie #4.



TYDZIEŃ 12

PLAN DZIEŃ PO DNIU

Dzień 1

Powtórka i plan

Wykonaj zadania 89–91. Wybierz tematy do powtórki i wypełnij kartę projektu.

90 min

Dzień 2

Klasy projektu

Wykonaj zadanie 92. Uruchom program, nawet jeśli na razie niewiele robi.

60 min

Dzień 3

Menu i logika

Wykonaj zadanie 93. Sprawdź każdą opcję menu.

60 min

Dzień 4

Pliki i statystyki

Wykonaj zadania 94–95.

90 min

Dzień 5

Testy i finał

Wykonaj zadanie 96 i wypełnij ostatnią kontrolę postępów.

60 min



Weekend: pokaż swój program komuś bliskiemu i opowiedz, jak działa. Należy Ci się nagroda!

TYDZIEŃ 12

PUŁAPKI I WSKAZÓWKI

Błąd

Cały program w metodzie main.

Poprawka

Osobne klasy i krótkie metody.

Kod podzielony na części łatwiej czytać, testować i rozbudowywać.

Błąd

Wszystkie funkcje napisane naraz.

Poprawka

Jedna funkcja, jedno uruchomienie.

Po każdej małej zmianie uruchom program. Wtedy od razu wiesz, który fragment spowodował błąd.

Błąd

Brak kopii działającej wersji.

Poprawka

Zapisuj kolejne wersje projektu.

Gdy coś zepsujesz, wrócisz do wersji, która działała. Z czasem poznaj system Git.

Wskazówka

Projekt nie musi być idealny. Lepszy prosty program, który działa, niż wielki plan, który nigdy nie powstanie.

TYDZIEŃ 12

ZADANIA TYGODNIA (1/2)

Najpierw powtórka, potem plan. Kod projektu napiszesz dopiero w zadaniu 92.

ZADANIE 89 — ☐
Moja mapa powtórki

Przejrzyj kontrole postępów z tygodni 1–11. Wypisz trzy tematy z najniższą oceną i jeszcze raz przerób ich mapy.

ZADANIE 90 — ☐
Test bez podpowiedzi

Bez zaglądania do map napisz program, który za pomocą pętli, warunku i własnej metody wypisze liczby pierwsze od 1 do 50.

ZADANIE 91 — ☐
Karta projektu

Wypełnij kartę projektu ze strony 85 dla Menedżera wydatków: cel, lista funkcji, klasy i plan na kolejne dni.

ZADANIE 92 — ☐
Klasy projektu

Utwórz klasę Wydatek z polami opis, kwota, kategoria i data oraz klasę Budzet, która przechowuje listę wydatków.

Jeśli zadanie 90 sprawi Ci kłopot, dopisz ten temat do listy z zadania 89.

TYDZIEŃ 12

ZADANIA TYGODNIA (2/2)

93 ZADANIE 93 — Menu główne

Dodaj pętlę z menu: dodaj wydatek, pokaż wszystkie, pokaż sumę, zakończ. Każdą opcję umieść w osobnej metodzie.

94 ZADANIE 94 — Zapis i odczyt

Przy wyjściu z programu zapisuj wydatki do pliku, a przy starcie wczytuj je z powrotem.

95 ZADANIE 95 — Statystyki wydatków

Za pomocą Streamów wypisz sumę wydatków w każdej kategorii oraz największy pojedynczy wydatek.

96 ZADANIE 96 — Odporność na błędy ★

Obsłuż błędne dane: tekst zamiast kwoty, kwotę ujemną i brak pliku. Program nie może przerwać pracy z powodu błędu.

Wskazówka

Po zadaniu 96 masz kompletną aplikację: klasy, *kolekcje*, pliki, wyjątki i Streamy w jednym programie.

TYDZIEŃ 12

KONTROLA POSTĘPÓW

Potrafię...

- [ ] zaplanować projekt przed pisaniem kodu
- [ ] podzielić program na klasy i metody
- [ ] połączyć kolekcje, pliki i wyjątki
- [ ] przetestować program na błędnych danych

Dziennik nauki

Czas nauki w tym tygodniu: _____ h

Z czego jestem najbardziej dumny lub dumna? _____



Podsumowanie etapu 4

Za Tobą dwanaście tygodni i 96 zadań. Znasz dziedziczenie, interfejsy, wyjątki, pliki i Streamy, a Twój pierwszy projekt działa. Już nie zaczynasz od zera.

Projekt gotowy, gdy...

- [ ] wszystkie opcje menu działają
- [ ] dane zostają po ponownym uruchomieniu
- [ ] błędne dane nie przerywają programu

Jak oceniasz ten tydzień?

1 2 3 4 5

TABLICA POSTĘPÓW: 12 TYGODNI

Zaznaczaj kolejne kroki i zamaluj tyle kółek, ile punktów dałeś lub dałaś sobie w kontroli postępów.

Tydzień 01
- [ ] mapy
- [ ] zadania 01–08
- [ ] kontrola

○ ○ ○ ○ ○

Tydzień 02
- [ ] mapy
- [ ] zadania 09–16
- [ ] kontrola

○ ○ ○ ○ ○

Tydzień 03
- [ ] mapy
- [ ] zadania 17–24
- [ ] kontrola

○ ○ ○ ○ ○

Tydzień 04
- [ ] mapy
- [ ] zadania 25–32
- [ ] kontrola

○ ○ ○ ○ ○

Tydzień 05
- [ ] mapy
- [ ] zadania 33–40
- [ ] kontrola

○ ○ ○ ○ ○

Tydzień 06
- [ ] mapy
- [ ] zadania 41–48
- [ ] kontrola

○ ○ ○ ○ ○

Tydzień 07
- [ ] mapy
- [ ] zadania 49–56
- [ ] kontrola

○ ○ ○ ○ ○

Tydzień 08
- [ ] mapy
- [ ] zadania 57–64
- [ ] kontrola

○ ○ ○ ○ ○

Tydzień 09
- [ ] mapy
- [ ] zadania 65–72
- [ ] kontrola

○ ○ ○ ○ ○

Tydzień 10
- [ ] mapy
- [ ] zadania 73–80
- [ ] kontrola

○ ○ ○ ○ ○

Tydzień 11
- [ ] mapy
- [ ] zadania 81–88
- [ ] kontrola

○ ○ ○ ○ ○

Tydzień 12
- [ ] mapy
- [ ] zadania 89–96
- [ ] kontrola

○ ○ ○ ○ ○

Cała tablica zaznaczona? To 96 zadań i 12 tygodni regularnej nauki.

OCENA KOŃCOWA

Oceń, jak dobrze znasz każdy obszar: 1 to „dopiero zaczynam”, 5 to „umiem to wyjaśnić innym”.

	1	2	3	4	5
Instalacja, środowisko, składnia, wejście i wyjście danych	○	○	○	○	○
Zmienne, typy danych, operatory i konwersje	○	○	○	○	○
Instrukcje warunkowe i rozwiązywanie problemów	○	○	○	○	○
Pętle for, while i do-while, break i continue	○	○	○	○	○
Metody, parametry, przeciążanie i pakiety	○	○	○	○	○
Tablice, Stringi, ArrayList i Collections	○	○	○	○	○
Wyjątki, pliki, interfejsy, generyki, lambdy i Streamy	○	○	○	○	○
OOP: klasy, obiekty, dziedziczenie, enkapsulacja i polimorfizm	○	○	○	○	○

Moje mocne strony

_____

_____

Do powtórki

_____

_____

KARTA TWOJEGO PROJEKTU



Wypełnij tę kartę przed każdym nowym projektem. Dziesięć minut planowania oszczędza godziny poprawiania kodu.

Nazwa projektu _____

Cel: co program ma robić?

_____

_____

Najważniejsze funkcje

1. _____
2. _____
3. _____

Klasy i ich zadania

_____

_____

Gdzie zapiszę dane? _____

Plan pracy

- ☐ Dzień 1: _____
- ☐ Dzień 2: _____
- ☐ Dzień 3: _____
- ☐ Dzień 4: _____
- ☐ Dzień 5: _____

CO DALEJ PO 12 TYGODNIACH



Plan się kończy, ale nauka trwa. Wybierz jedną ścieżkę i trzymaj się rytmu, który już masz.



Wracaj do map

Raz w tygodniu przejrzyj mapy z tematów, które w ocenie końcowej dostały 1–3 punkty.



Ćwicz z wyzwaniami

Rozwiązuj wyzwania z Bonusu #2, po jednym dziennie, od najłatwiejszych do najtrudniejszych.



Buduj projekty

Zrób kolejne projekty z Bonusu #4. Każdy zacznij od karty projektu ze strony 85.



Pogłębiaj OOP

Wróć do Bonusu #5 i przebuduj swój Menedżer wydatków tak, aby lepiej korzystał z interfejsów.

Następne tematy

Git i GitHub • testy JUnit • bazy danych SQL • Maven lub Gradle



GRATULACJE!

Masz za sobą całą drogę: od instalacji JDK do własnego projektu. Dwanaście tygodni regularnej nauki, 96 wykonanych zadań i program, który naprawdę działa.

Pamiętaj: każdy programista zaczynał od pierwszego „Cześć, Java!”. Ty masz to już dawno za sobą.

Nie bój się kodu. Po prostu pisz dalej.